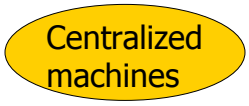
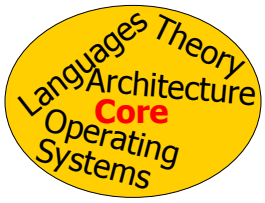
Distributed Sensor and Cyber-Physical Systems

Tarek Abdelzaher
Department of Computer Science
University of Illinois at Urbana Champaign

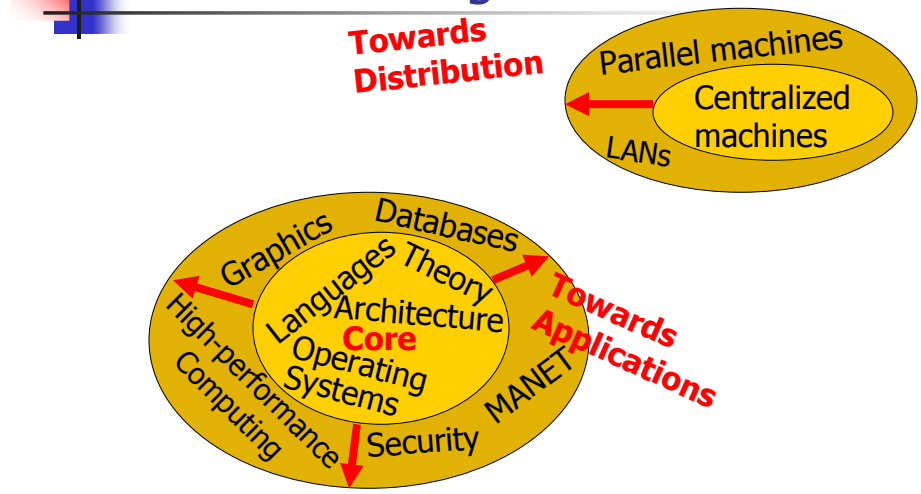
Where is Computer Science Research Going?

The beginning:

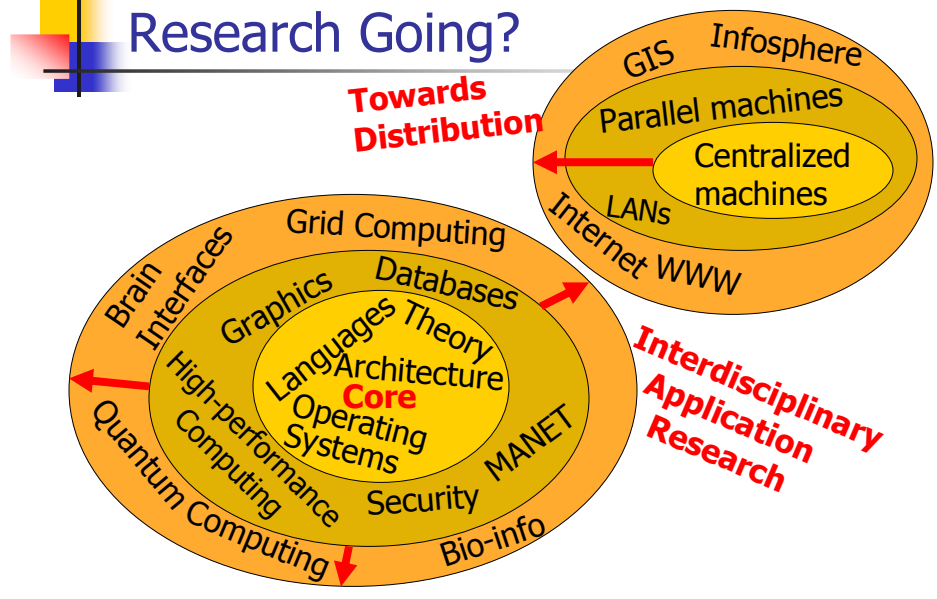


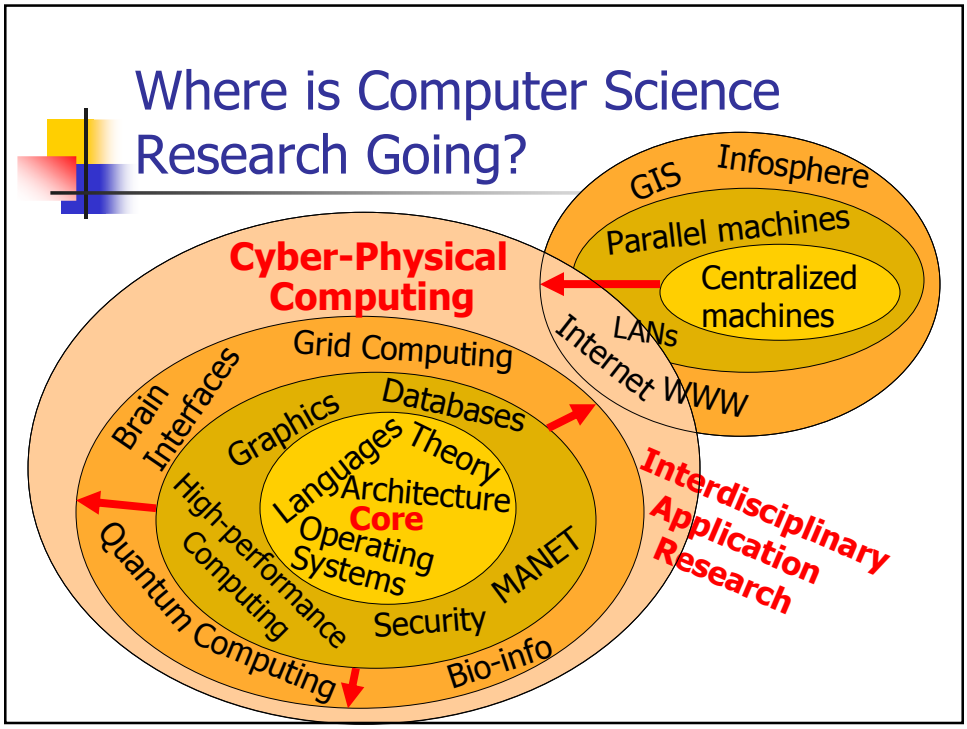


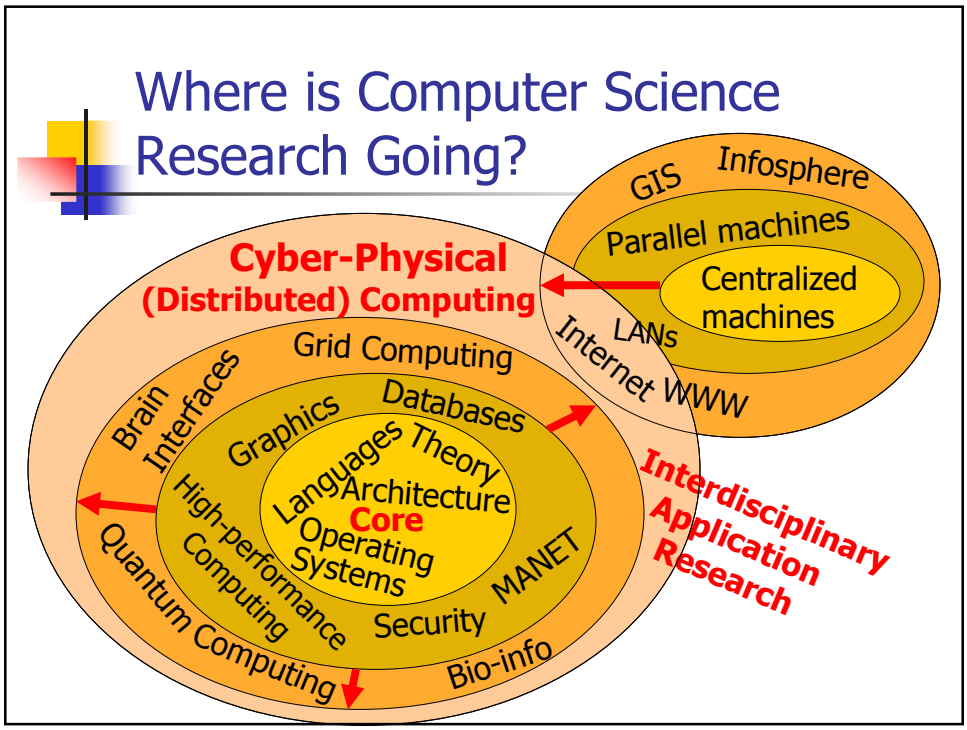
Where is Computer Science Research Going?



Where is Computer Science Research Going?




Where is Computer Science Research Going?
Cyber-Physical Computing
GIS
Infosphere
Parallel machines
Centralized machines
LANs
Internet WWW
Grid Computing
Brain Interfaces
Graphics
Databases
Languages
Theory
Architecture
Core
Operating Systems
High-performance Computing
MANET
Security
Quantum Computing
Bio-info
Interdisciplinary Application Research


Where is Computer Science Research Going?
Cyber-Physical (Distributed) Computing
GIS
Infosphere
Parallel machines
Centralized machines
LANs
Internet WWW
Grid Computing
Brain Interfaces
Graphics
Databases
Languages
Theory
Architecture
Core
Operating Systems
High-performance Computing
MANET
Security
Quantum Computing
Bio-info
Interdisciplinary Application Research

Where is Computer Science Research Going?

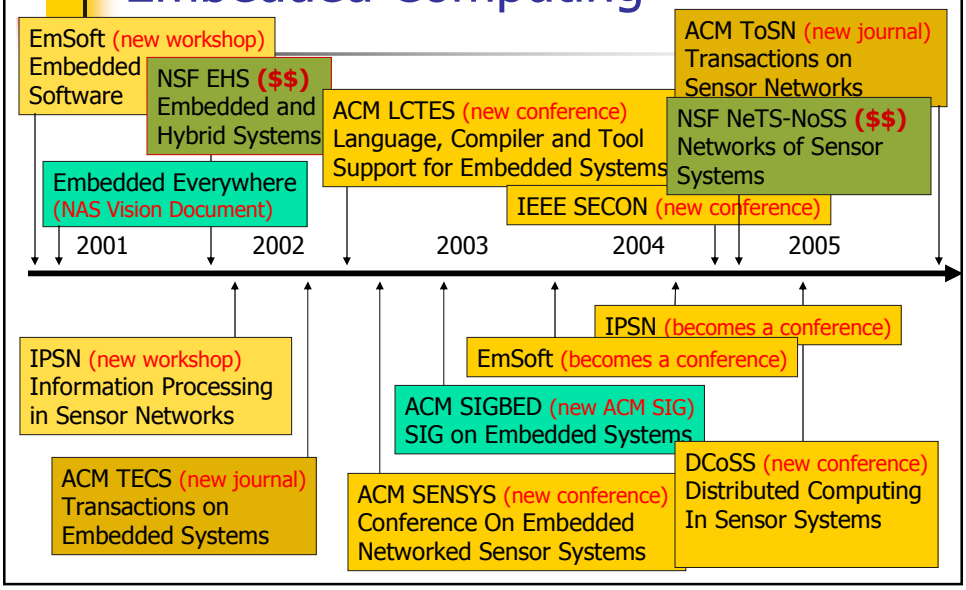
Cyber-Physical Distributed Systems

For example:

In the US, the **Presidential Counsel of Advisors in Science and Technology** named systems that interact with the physical world the

#1 Research Priority in the US

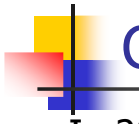
Expanding Interest in Deeply Embedded Computing



Distributed Cyber-Physical Systems versus Embedded Systems?

- A more holistic/interdisciplinary approach
- More distribution
- Less structure
- Interactions between multiple autonomous domains (privacy and security implications)
- Increasingly open systems

→ Think: Distributed versus parallel computing



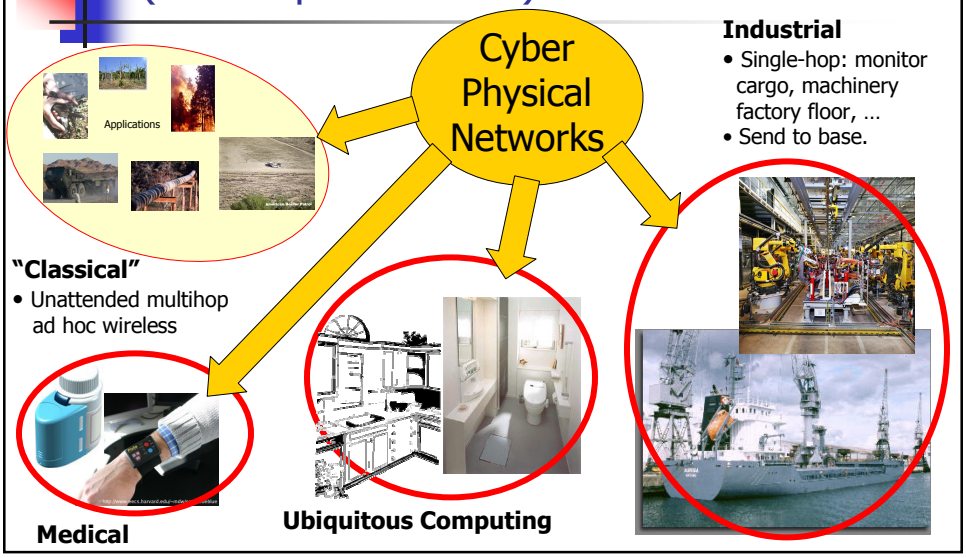
Cyber-Physical Systems

- In 2007 and 2008:
  - PCAST Report on US Competitiveness names CPS as #1 priority
  - RTSS (#1 Real-time conference) adds CPS track
  - IPSN/HSCC/RTAS 2008 Join for CPS Week
  - Several New CPS Workshops
    - High Confidence Cyber-Physical Systems (July 2007)
    - From Embedded Systems to CPS (w/ RTAS 2008)
    - First International Workshop on CPS (w/ ICDCS 2008)
    - …
  - NSF Initiative to fund CPS track

Publication Counts

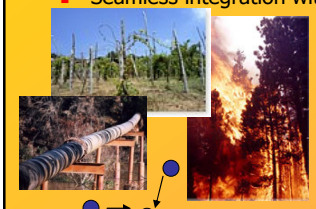
Year	Sensor Networks	Cyber-Physical	Internet	
1997	6			
1998	13		1988	
1999	21		1989	71
2000	39		1990	110
2001	95		1991	132
2002	213		1992	200
2003	609		1993	359
2004	1489		1994	863
2005	2423		1995	1711
2006	3365	1	1996	3954
		4	1997	6207
			1998	6657
			1999	7017
			2000	8831
			2001	10224
			2002	12755

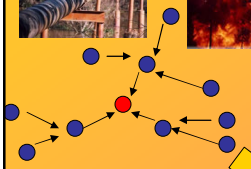
Force #1: Moore's Law
(Device proliferation)



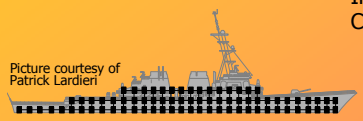
Force #2: Integration at Scale
(Isolation has cost!)

- Low end: ubiquitous embedded devices
  - Large-scale networked embedded systems
  - Seamless integration with a physical environment
- High end: complex systems with global integration
  - Examples: Global Information Grid, Total Ship Computing Environment





Picture courtesy of Patrick Lardieri



Total Ship Computing Environment (TSCE)

Global Information Grid

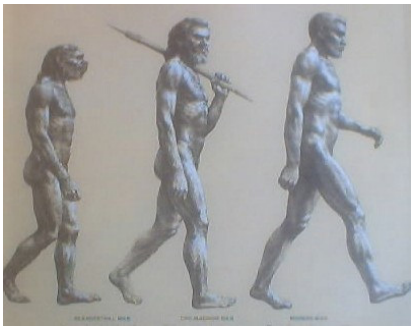
World Wide Sensor Web (Feng Zhao)

Integration and Scaling Challenges

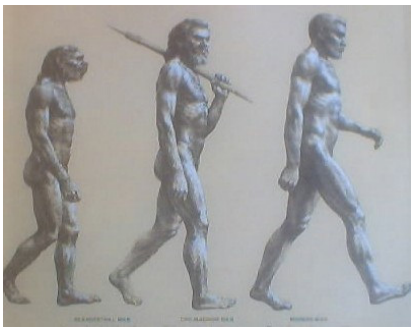
Future Combat System (Rob Gold)

Low End — High End

Force #3: Biological Evolution



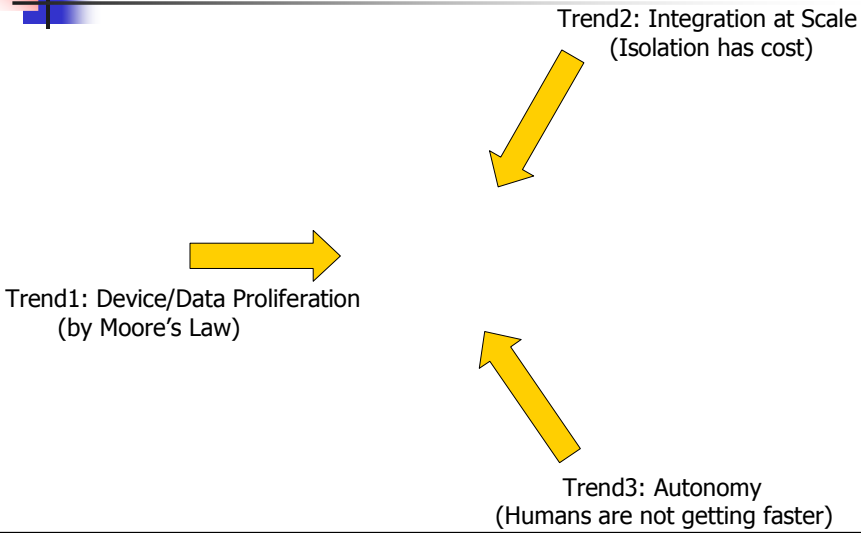
Force #3: Biological Evolution



- **It's too slow!**
  - The exponential proliferation of data sources (afforded by Moore's Law) is ***not*** matched by a corresponding increase in human ability to consume information!

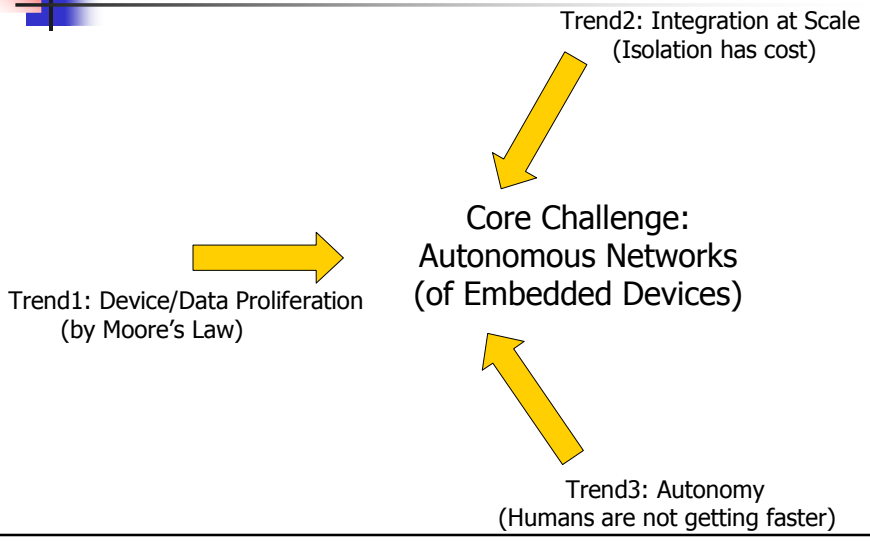
  → Increasing autonomy (human out of the loop), direct world access

Confluence of Trends
The Overarching Challenge

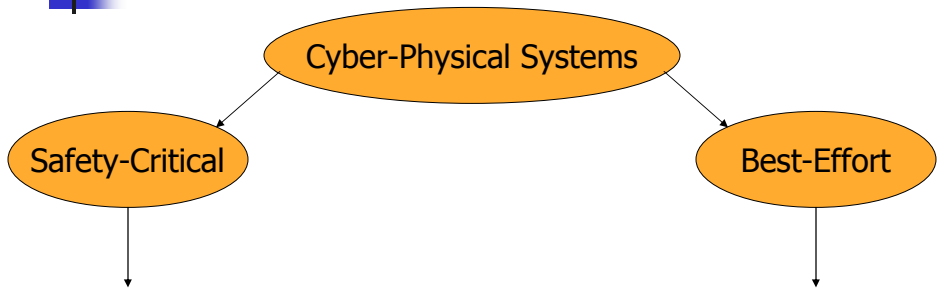


Confluence of Trends

The Overarching Challenge



The Distributed Cyber-Physical System Research Landscape



High-end systems (avionics, power grids)
Specification Languages
Model-driven development
Formal Verification
Safety-Properties
Design tools

Lower-end systems
Programming Languages
Architecture/abstractions
Troubleshooting tools
Adaptive-behavior
Design tools

This Class

- **Instructor**
  Tarek Abdelzaher, 4126 SC, Tel: (217) 265-6793, e-mail: zaher
- **Lecture Times**
  Schedule: Wednesdays and Fridays: 2:00-3:15pm, 1131 SC
- **Web page**
  http://www.cs.uiuc.edu/class/cs598tar

Primarily Paper Reading

- Check class webpage for reading list soon (no reading due this week).
- Divide into groups of up to 3 people (partner on projects and homework).
- Each group collectively will be required to read two to four papers per week (from a reading list) on the topic covered in class, and type a half-page to one page summary of the paper indicating:
  - The main contributions of the paper.
  - A critique (positive and/or negative) of this paper.
  - The points of strength (best things you liked about the paper).
  - The points of weakness (things you didn't like about the paper).
  - Opportunities for future work on the topic.
- Summaries of assigned research papers (one per group) are to be submitted to the instructor by e-mail by 9pm of the day before class. (Please include the words "598 SUMMARY" in the subject of the e-mail, and include the group members and title of the critiqued paper in the body.
- A discussion of the research topic will ensue in class.

Grading

- **10%:** Class participation, and discussion of research papers.
- **10%:** Group summaries of the papers covered (all group members share the same grade). More credit will be given to groups with creative and original opinions, and on their ability to defend their correctness.
- **15%:** Open-book take-home midterm (out 3/14)
- **15%:** Open-book take-home final (out 5/2)
- **50%:** Substantial group course project.

Project

- **ASAP:** Find partners (a group of 3 is best)
- **2/8:** Submit project proposal (1 page)
- **3/12-3/14:** In class work-in-Progress presentation
- **4/25-4/30:** Final in-class presentation
- **5/2:** Final written report
- **Week of finals:** Project demo and discussion.

- Access will be provided to a project testbed in the second half of the semester.

Confluence of Trends

The Overarching Challenge

Trend2: Integration at Scale
(Isolation has cost)

Core Challenge:
Autonomous Networks
(of Embedded Devices)

Trend1: Device/Data Proliferation
(by Moore's Law)

Trend3: Autonomy
(Humans are not getting faster)

Sensor "LANs"

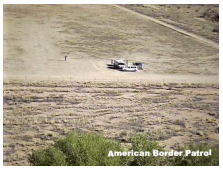
Habitat Monitoring

Precision Agriculture

Disaster Response

- Constrained deployment
- Distribution at scale
- Interaction with a physical environment
- Unattended operation

Application Examples



Border Control

Miniaturization

CPS Networks

- Integration of
  - computation,
  - communication, and
  - interaction with the physical world

CPS Networks

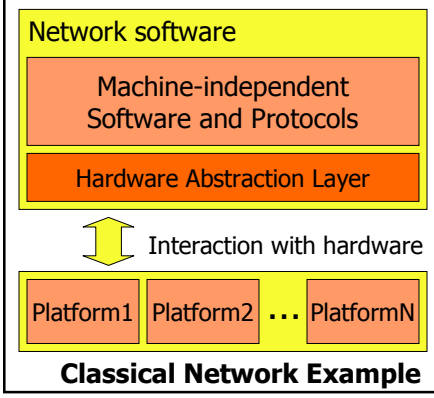
- Integration of
  - computation,
  - communication, and
  - interaction with the physical world

Network Challenge – The "Adaptation" Layer

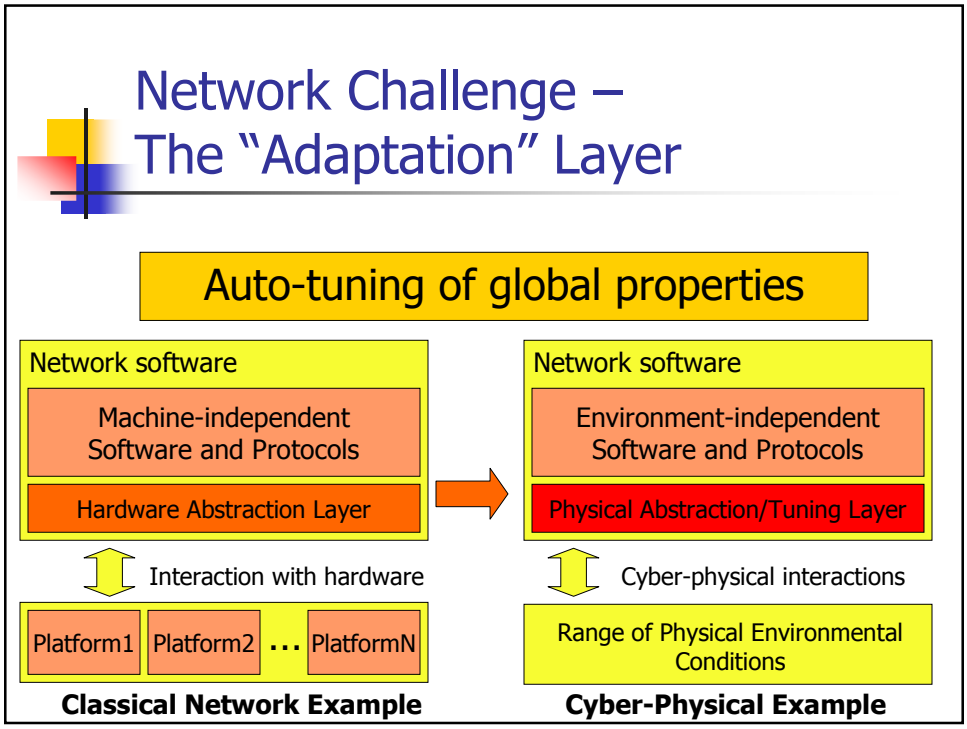
Auto-tuning of global properties

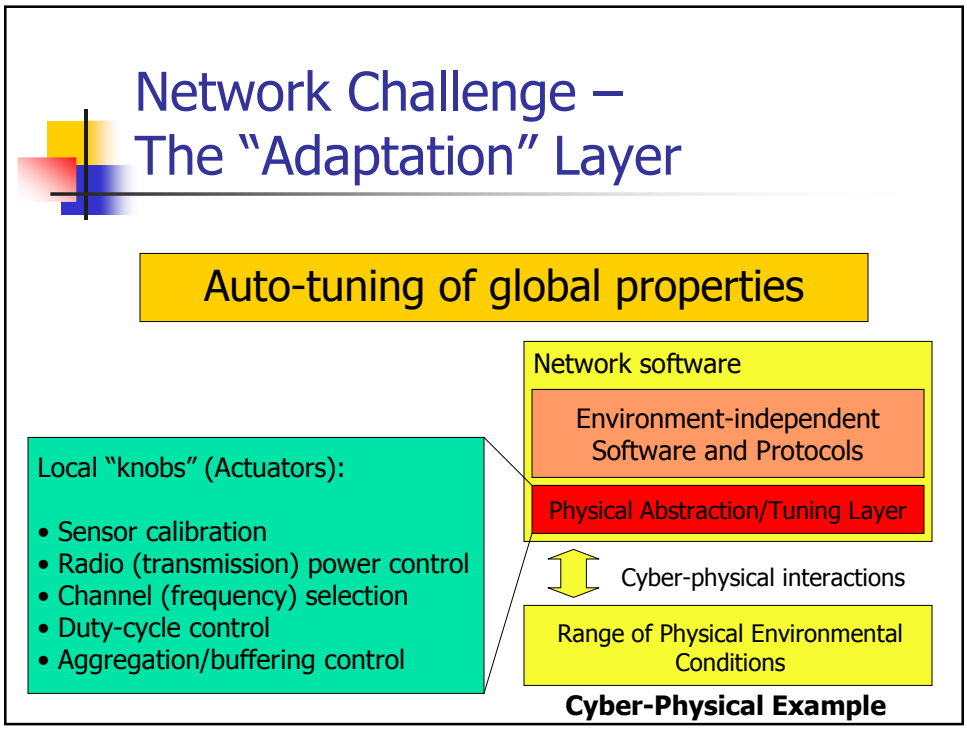
Network Challenge – The "Adaptation" Layer

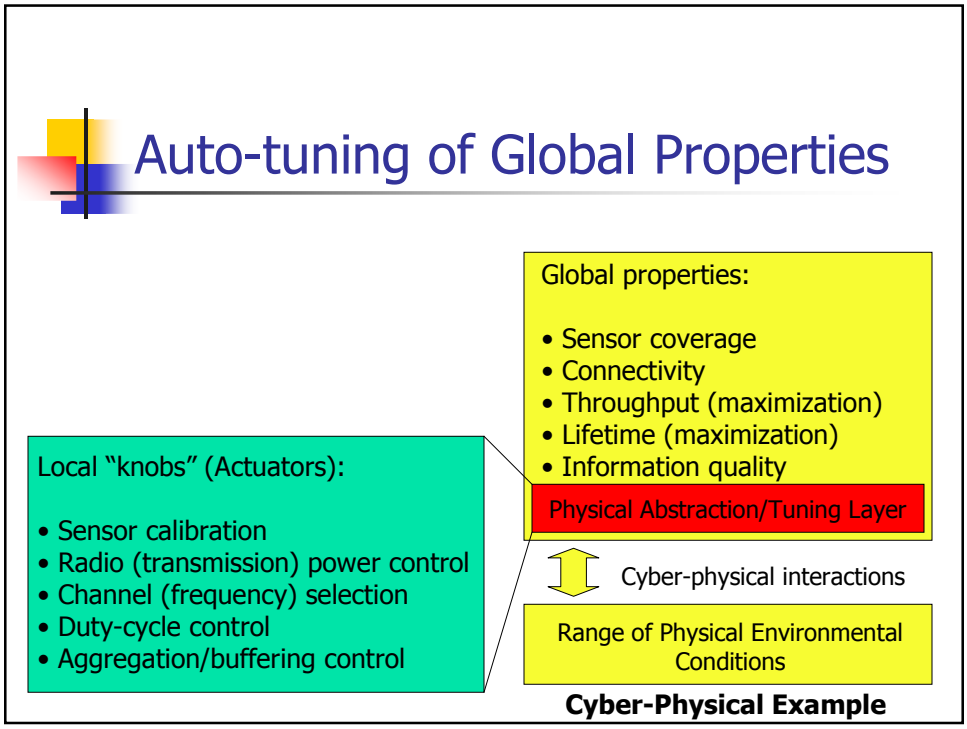
Auto-tuning of global properties

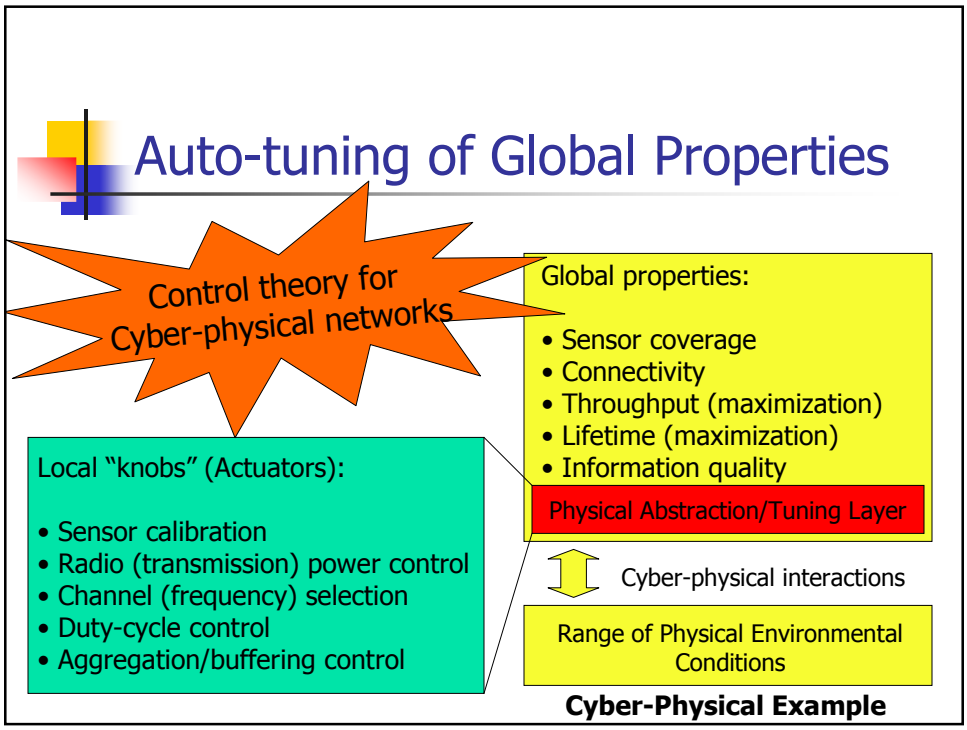


Classical Network Example

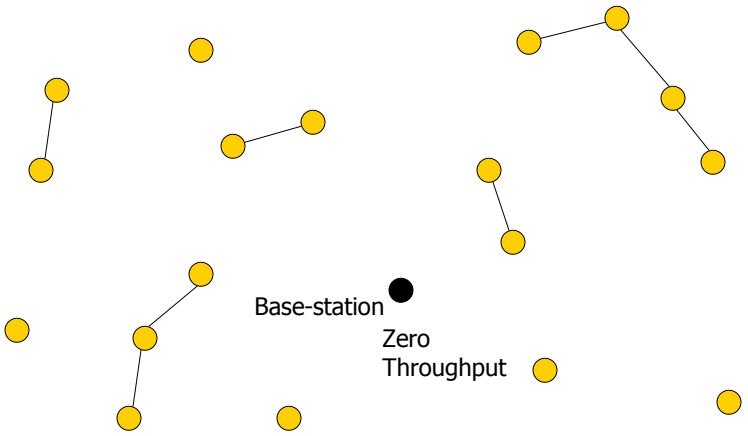

Network Challenge –
The "Adaptation" Layer
Auto-tuning of global properties
Network software
Machine-independent Software and Protocols
Hardware Abstraction Layer
Interaction with hardware
Platform1
Platform2
...
PlatformN
Classical Network Example
Network software
Environment-independent Software and Protocols
Physical Abstraction/Tuning Layer
Cyber-physical interactions
Range of Physical Environmental Conditions
Cyber-Physical Example


Network Challenge –
The "Adaptation" Layer
Auto-tuning of global properties
Local "knobs" (Actuators):
• Sensor calibration
• Radio (transmission) power control
• Channel (frequency) selection
• Duty-cycle control
• Aggregation/buffering control
Network software
Environment-independent Software and Protocols
Physical Abstraction/Tuning Layer
Cyber-physical interactions
Range of Physical Environmental Conditions
Cyber-Physical Example


Auto-tuning of Global Properties
Global properties:
• Sensor coverage
• Connectivity
• Throughput (maximization)
• Lifetime (maximization)
• Information quality
Physical Abstraction/Tuning Layer
Local "knobs" (Actuators):
• Sensor calibration
• Radio (transmission) power control
• Channel (frequency) selection
• Duty-cycle control
• Aggregation/buffering control
Cyber-physical interactions
Range of Physical Environmental Conditions
Cyber-Physical Example

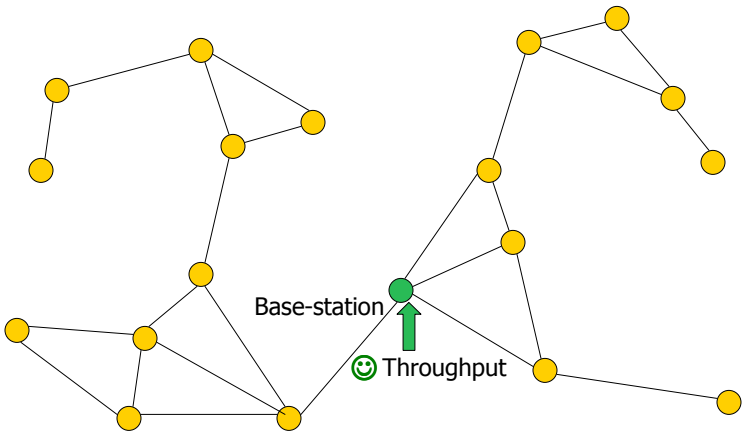

Auto-tuning of Global Properties
Control theory for Cyber-physical networks
Global properties:
• Sensor coverage
• Connectivity
• Throughput (maximization)
• Lifetime (maximization)
• Information quality
Physical Abstraction/Tuning Layer
Local "knobs" (Actuators):
• Sensor calibration
• Radio (transmission) power control
• Channel (frequency) selection
• Duty-cycle control
• Aggregation/buffering control
Cyber-physical interactions
Range of Physical Environmental Conditions
Cyber-Physical Example

Classical Example:
Power/Connectivity Control (for Max. Throughput)



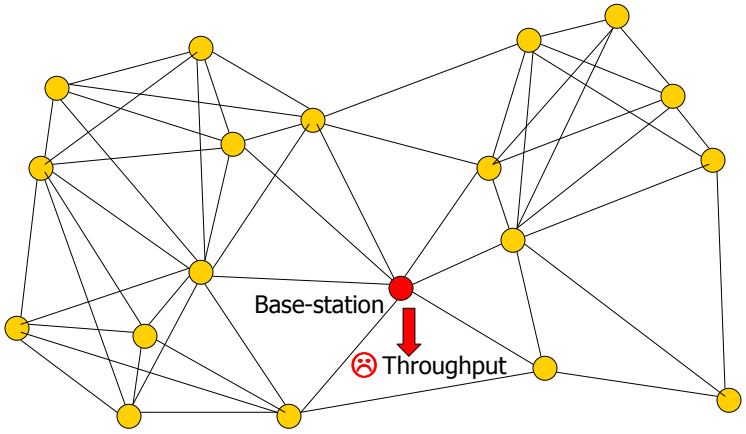
Transmission range too short (partitions)

Classical Example:
Power/Connectivity Control



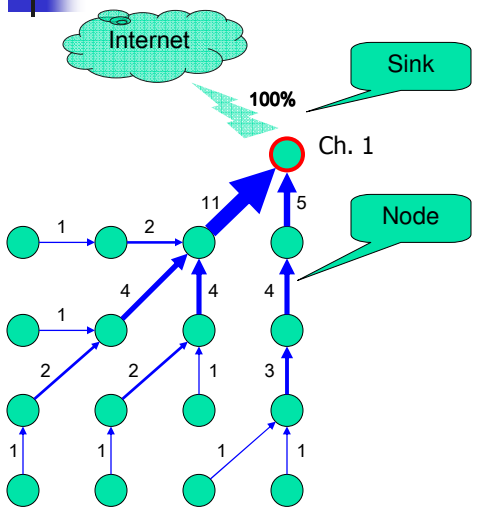
Just right

Classical Example:
Power/Connectivity Control



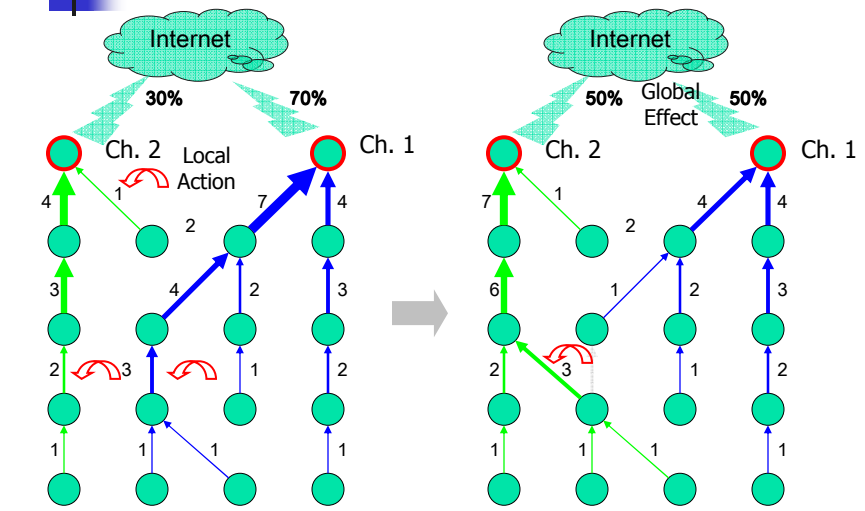
Transmission range too long (collisions)

Load Balancing Example:
Channel Allocation Control

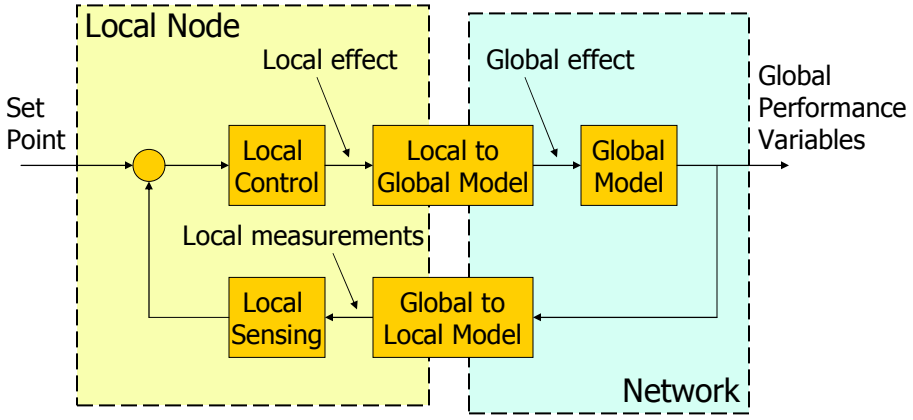


- Nodes form a collection tree rooted at the sink
- Nodes both generate and forward data toward the sink
- Collision and interference get worse closer to the sink

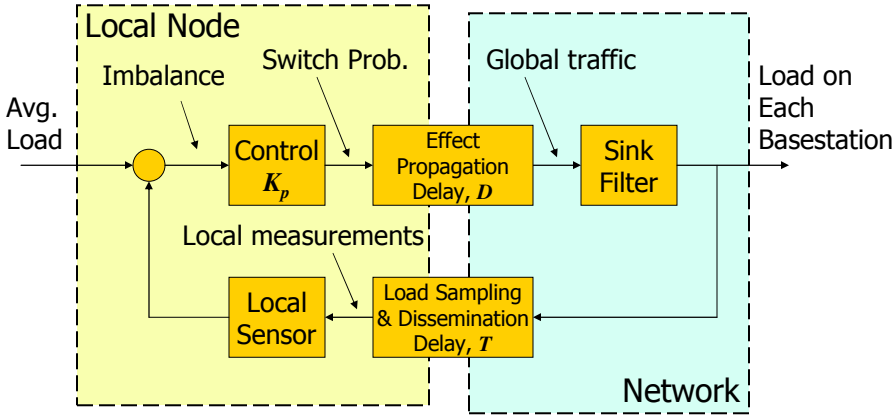
Load Balancing Example:
Channel Allocation Control



A Simple Control-Theoretic Model of Adaptive Behavior

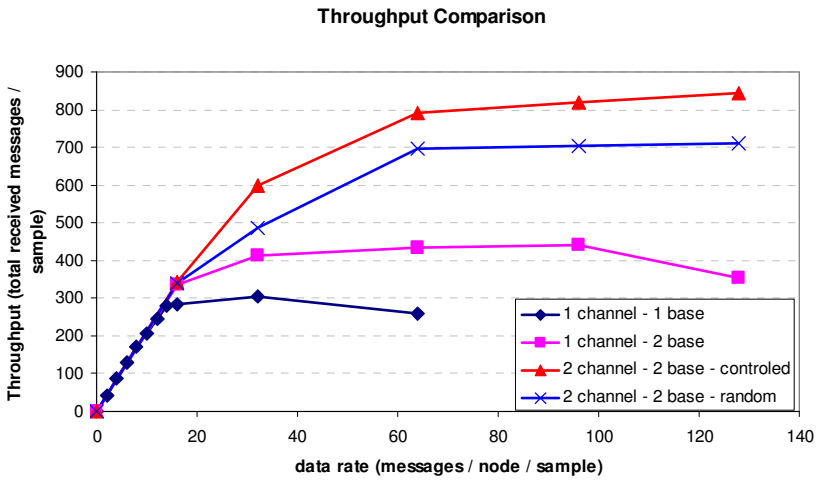


A Simple Control-Theoretic Model of Adaptive Behavior



Example: Control Testbed

Throughput Comparison with and without Automatic Control



General Challenge: Control of Global Network Behavior

- Sensor network protocols are localized; nodes act independently, locally, in response to local stimuli.
- How to argue about the global effects of such localized protocols when performed by all nodes?
- How to induce and analyze convergence to desired global properties?
- How to incorporate end-to-end time and total capacity constraints?
- Inspirations from control theory (convergence, stability), Markov decision theory (Markov chains, stochastic models), biology (behavior of social insects, swarm intelligence, bio-differentiation), physics (phase transitions, crystallization), …

CPS Networks

- Integration of
  - computation,
  - communication, and
  - interaction with the physical world

CPS Networks

- Integration of
  - computation,
  - communication, and
  - interaction with the physical world

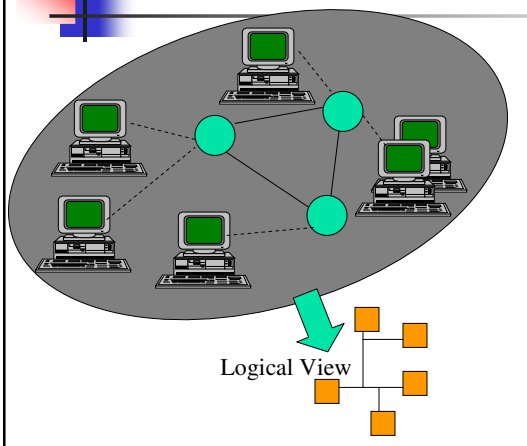
Sensor Networks

- Integration of
  - computation,
  - communication, and
  - interaction with the physical world

New Programming Paradigms

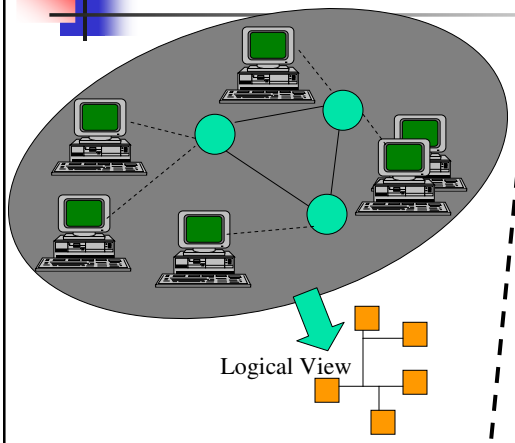
Distributed Troubleshooting

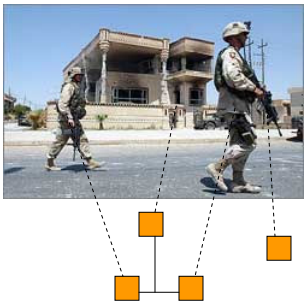
Challenge: Distributed Programming Abstractions



- Distributed programming paradigms
  - Abstract distributed communication
  - Provide location transparency

Challenge: Distributed Programming Abstractions

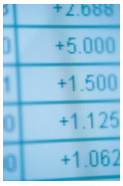


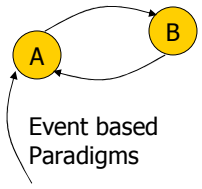


- Distributed programming paradigms
  - Abstract distributed communication
  - Provide location transparency
- Sensor Network Programming Abstractions
  - Represent the physical world to the programmer
  - Abstract distributed interaction with the physical environment

Architectures/Paradigms for In-network Computation

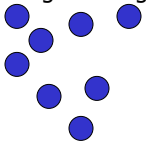
Sensor network as a Database

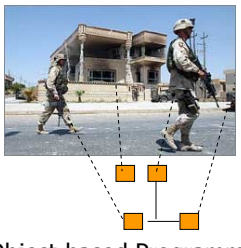




Event based Paradigms

Node-based Programming



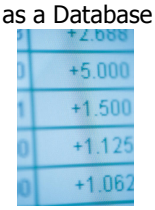


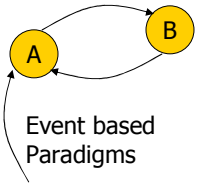
Object-based Programming

- Different paradigms for different applications
- Each supported by a black-box runtime system with a paradigm-specific API
- Can't mix and match

Architectures/Paradigms for In-network Computation

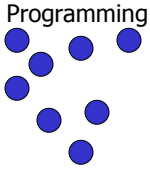
Sensor network as a Database

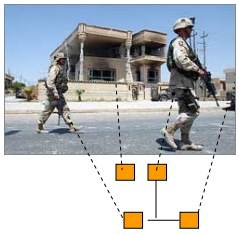




Event based Paradigms

Node-based Programming





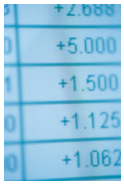
Object-based Programming

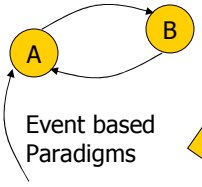
- Contrast with networking
  - Layered abstractions
  - Different programmers can operate together at different layers
  - Programs can use APIs from multiple layers

Layer 5
Layer 4
Layer 3
Layer 2
Layer 1

Layered (Network Protocols and) Programming Abstractions?

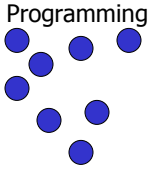
Sensor network as a Database

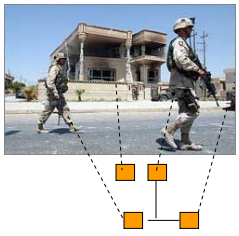




Event based Paradigms

Node-based Programming





Object-based Programming

- Towards a layered approach to programming abstractions?

?

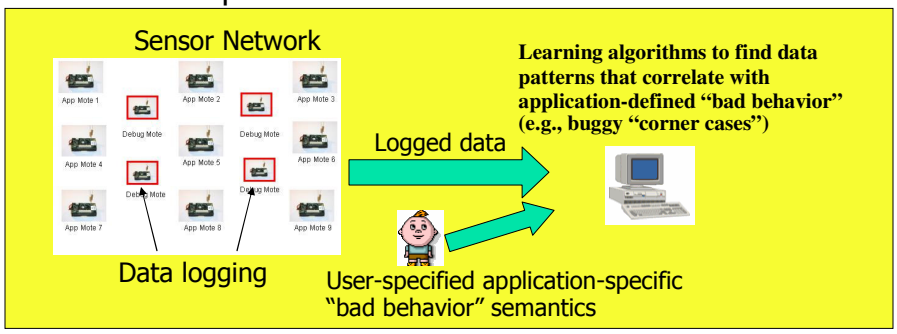
Layer 5
Layer 4
Layer 3
Layer 2
Layer 1

Challenge: Network Diagnosis and Troubleshooting Services

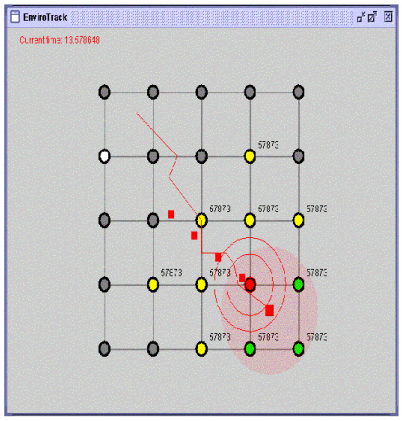
- Network performance problems and software bugs due of interaction among distributed components
  - Problems may not manifest themselves when testing individual components
  - Problems may appear to be non-repeatable

A Machine Learning Approach to Network Troubleshooting

- Monitor network state
- Label collected data into a "good pile" and a "bad pile" as defined by the user
- Learn data patterns that correlate with bad behaviour.



Case Study: Envirotrack

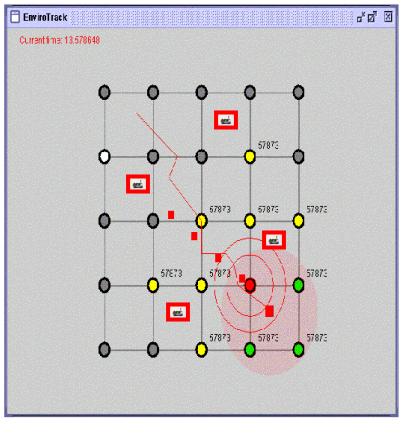


Envirotrack

- Distributed target tracking protocol (nodes that see target form group)
- Assigns unique ObjectId to each target
- Target movement causes leader handoff
- Sometimes hand-off failed resulting in more than one "detected target" for one physical target.
  - Why?

53

Case Study: Envirotrack



Envirotrack

- Distributed target tracking protocol (nodes that see target form group)
- Assigns unique ObjectId to each target
- Target movement causes leader handoff
- Sometimes hand-off failed resulting in more than one "detected target" for one physical target.
  - Why?

54

Generated Results

- Leader handoff fails in singleton groups
  - **Uncovered bug: Singleton groups are not addressed**
- Holes in sensing coverage with slow target
  - **Violated assumption: 100% sensing coverage**

55

A Future Network

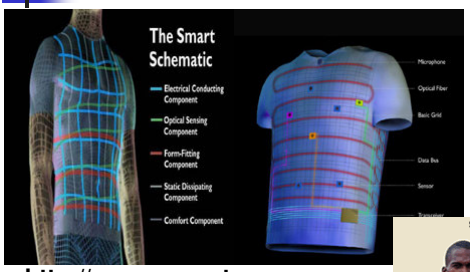
might look something like this …

A Future Network

might look something like this …

Human-Centric Applications

Empirical Evidence and Business case



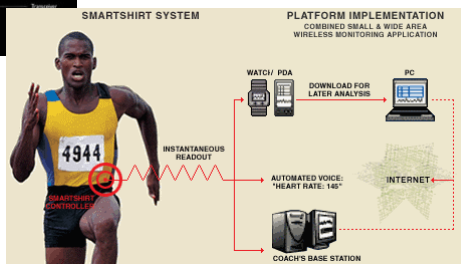
http://www.sensatex.com

The master controller

Nike -iPod

Spot

Wii



Technology introduced by 2007

Business Case?
Health and Wellness

Health Vault

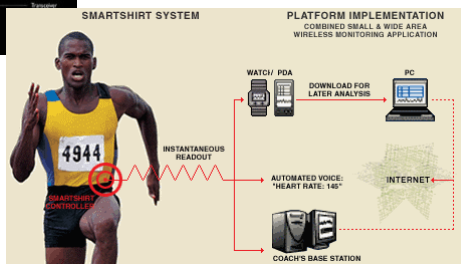
The master controller

http://www.sensatex.com

Nike -iPod

Spot

Wii



Business Case?
Sports and Entertainment

The master controller

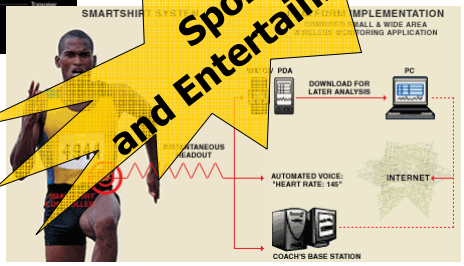
http://www.sensatex.com

iPod

Sports and Entertainment

Spot

Wii



Business Case?

Multiplayer Games (beyond 2nd life)

Health and Wellness

http://www.sensatex.com

The master controller

iPod

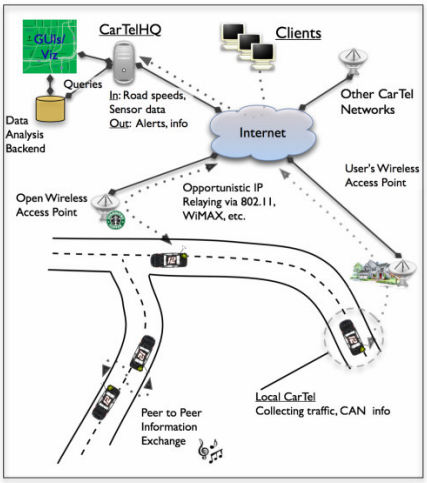
Sports and Entertainment

Spot

Games

Community Application Example

CarTel (Sam Madden)



Reprinted from http://cartel.csail.mit.edu/overview.html

- An ad hoc network of vehicles with sensors
- Can measure road congestion
- Can generate maps of road conditions, etc.
- Given uncontrolled mobility model, how to perform global data operations (query dissemination, collection, etc)?

Challenge: Privacy Preserving Statistics

- A community of clients, each has a private data set S_i
- An information distillation service needs to compute some aggregate community quantity Q from the data sets S_i, $Q = \mathrm{f}(S_1, S_2, \ldots)$
- The problem statement: Perturb the private set S_i to generate the shareable set S_i', such S_i cannot be recovered from S_i' but Q can still be (approximately) computed, $Q = \mathrm{f}(S_1', S_2', \ldots)$

Challenge: Opportunistic Mobility and Disruption Tolerance

- Opportunistic (symbiotic or parasitic) mobility
  - The common case: mobile sensors that do not control their own mobility but follow motion of host (e.g., a GPS in a car)
  - Data hitchhiking: data have a destination - sensors copy data to devices that are "going in the right direction"
  - Host should benefit (symbiotic)
  - Examples: CarTel, SkiScape, …
- External (phenomenon) mobility
  - Sensor mobility is determined by the measured object/phenomenon
  - Examples: cargo tracking, flow monitoring using floating sensors, etc.
- Scheduled mobility
  - Specialized device (robot, UAV, …) acts as a data mule for a collection of static sensors

Eventually:
A Network Paradigm Shift:
Communication → Information Distillation

- Present Internet

Goal:

Communication

- Connecting people
- Protocols geared primarily for point-to-point communication
- Data consumption applications: e-mail, VoIP, Web, P2P, …

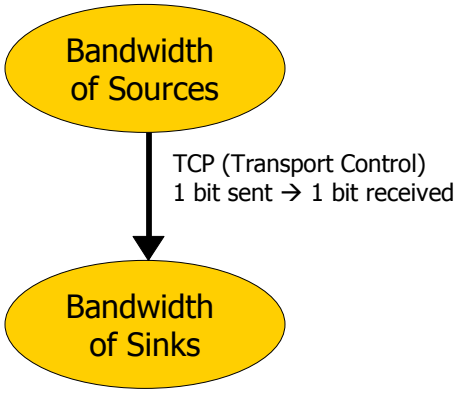
- Future Embedded Device Nets

Goal:

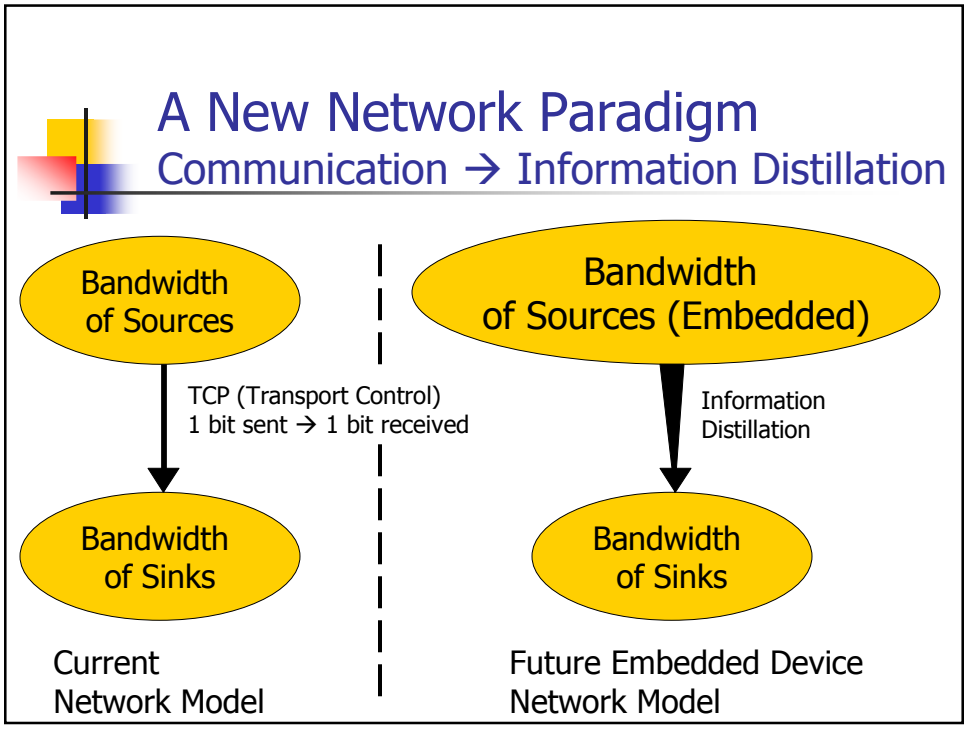
Information Distillation

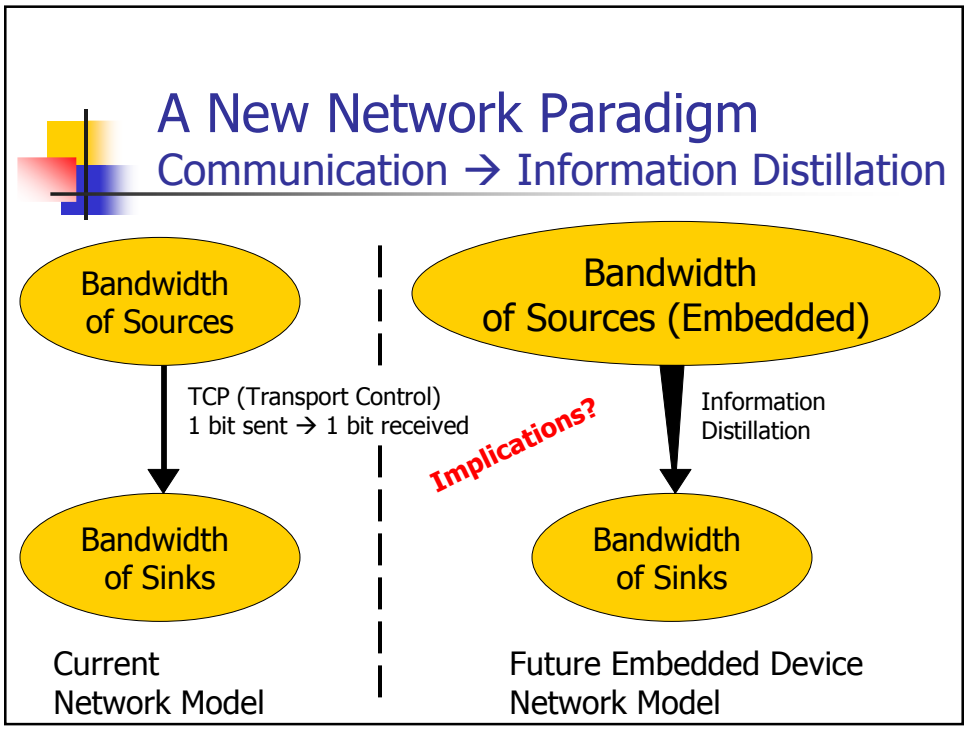
- Reducing data from sensing inputs to actionable information
  - Human is out of the main data reduction loop
- Protocols geared for data filtering, aggregation, mining, …
- Data reduction applications: Emergency detection/response, statistical community analysis, …

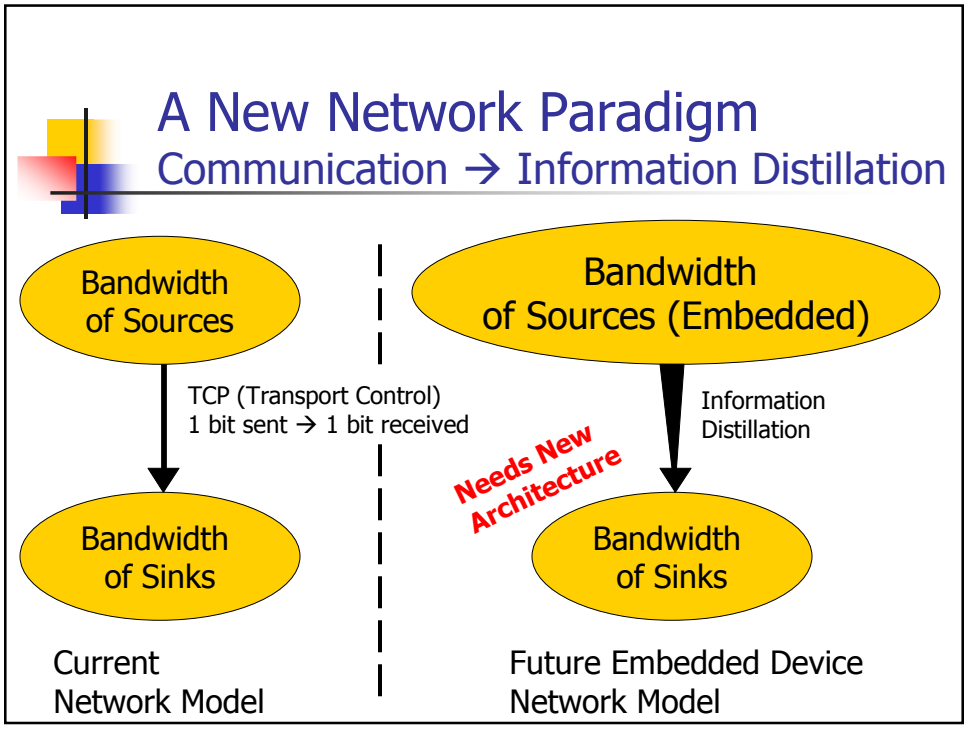
A New Network Paradigm
Communication → Information Distillation

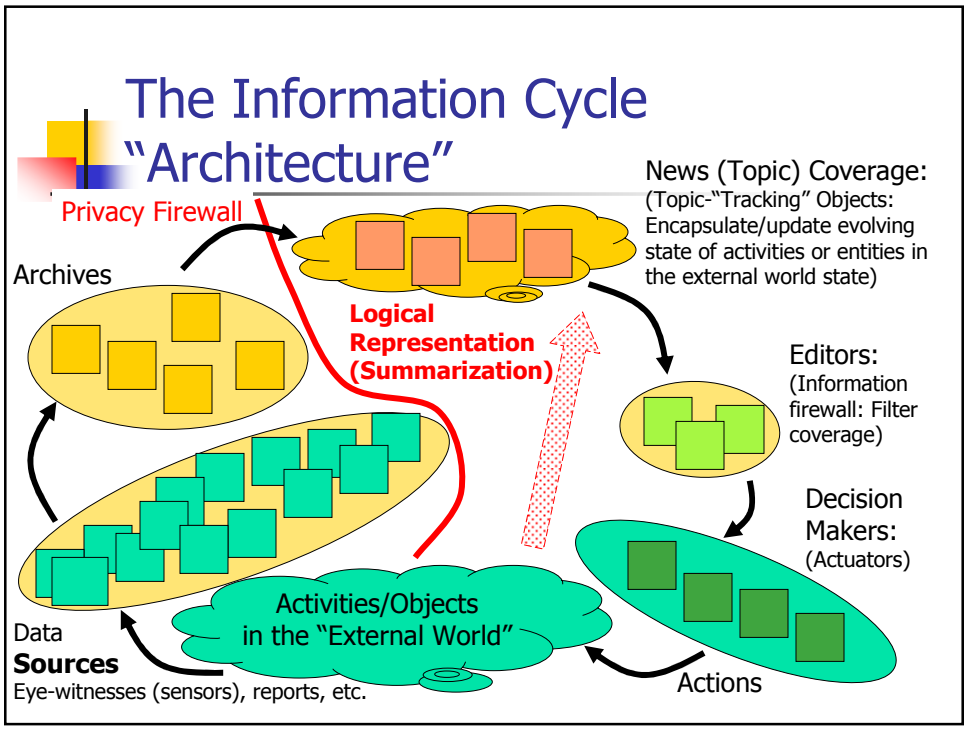


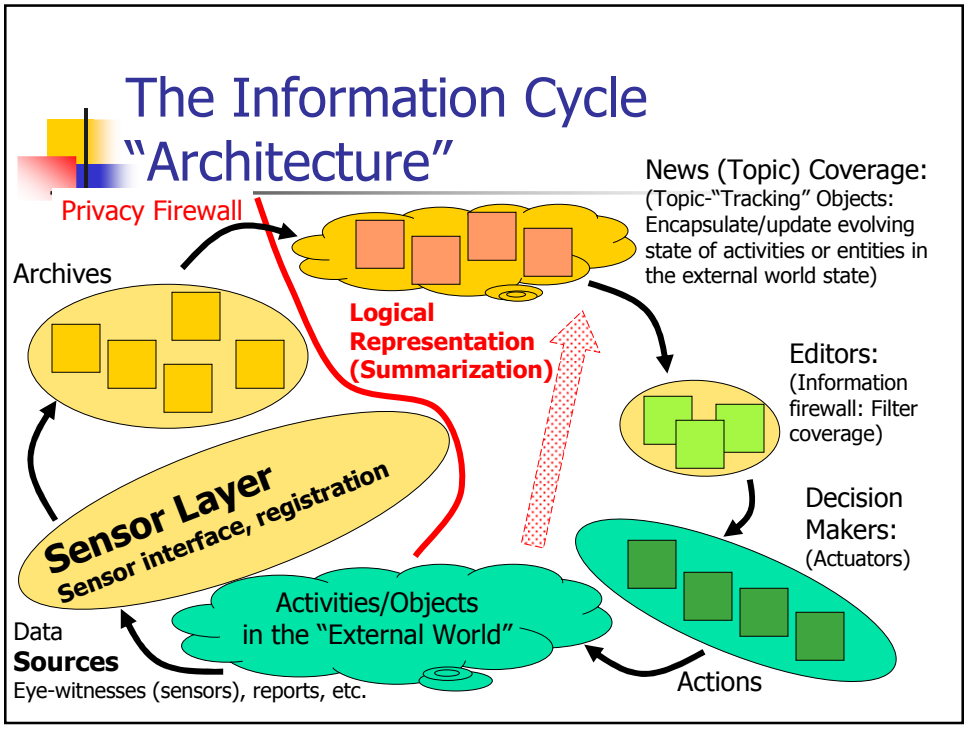
Current Network Model

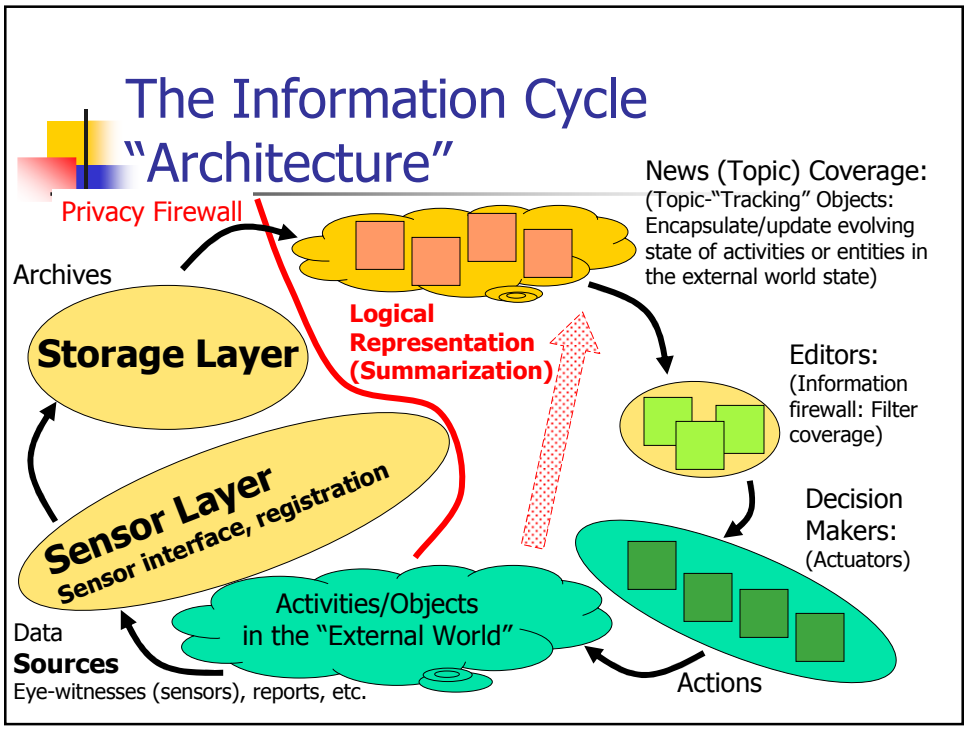

A New Network Paradigm
Communication → Information Distillation
Bandwidth of Sources
Bandwidth of Sources (Embedded)
TCP (Transport Control)
1 bit sent → 1 bit received
Information Distillation
Bandwidth of Sinks
Bandwidth of Sinks
Current Network Model
Future Embedded Device Network Model

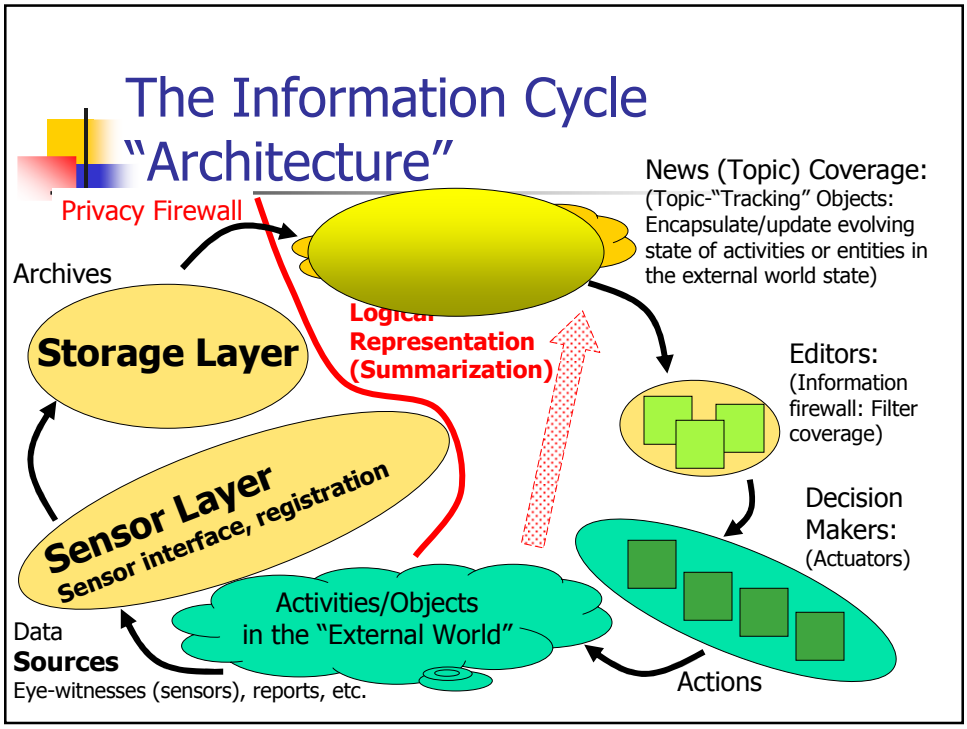

A New Network Paradigm
Communication → Information Distillation
Bandwidth of Sources
Bandwidth of Sources (Embedded)
TCP (Transport Control)
1 bit sent → 1 bit received
Implications?
Information Distillation
Bandwidth of Sinks
Bandwidth of Sinks
Current Network Model
Future Embedded Device Network Model

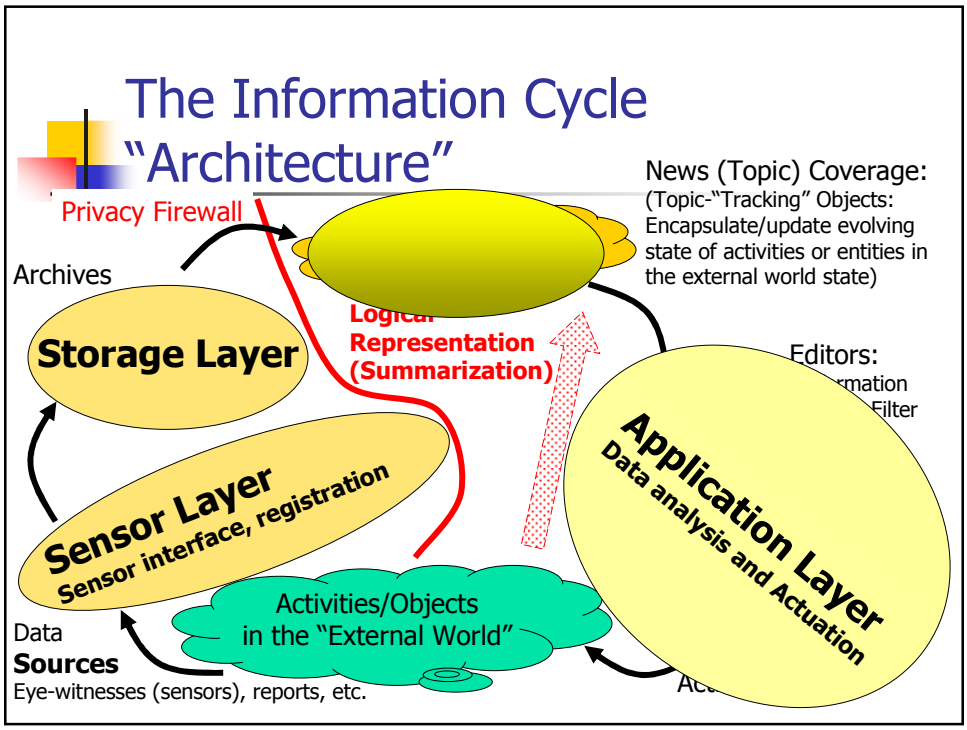

A New Network Paradigm
Communication → Information Distillation
Bandwidth of Sources
Bandwidth of Sources (Embedded)
TCP (Transport Control)
1 bit sent → 1 bit received
Information Distillation
Needs New Architecture
Bandwidth of Sinks
Bandwidth of Sinks
Current Network Model
Future Embedded Device Network Model


The Information Cycle "Architecture"
Privacy Firewall
News (Topic) Coverage:
(Topic-"Tracking" Objects: Encapsulate/update evolving state of activities or entities in the external world state)
Archives
Logical Representation (Summarization)
Editors:
(Information firewall: Filter coverage)
Decision Makers:
(Actuators)
Activities/Objects in the "External World"
Data Sources
Eye-witnesses (sensors), reports, etc.
Actions


The Information Cycle
"Architecture"
Privacy Firewall
Archives
News (Topic) Coverage:
(Topic-"Tracking" Objects:
Encapsulate/update evolving
state of activities or entities in
the external world state)
Logical
Representation
(Summarization)
Editors:
(Information
firewall: Filter
coverage)
Decision
Makers:
(Actuators)
Sensor Layer
Sensor interface, registration
Activities/Objects
in the "External World"
Data
Sources
Eye-witnesses (sensors), reports, etc.
Actions


The Information Cycle
"Architecture"
Privacy Firewall
Archives
News (Topic) Coverage:
(Topic-"Tracking" Objects:
Encapsulate/update evolving
state of activities or entities in
the external world state)
Storage Layer
Logical
Representation
(Summarization)
Editors:
(Information
firewall: Filter
coverage)
Decision
Makers:
(Actuators)
Sensor Layer
Sensor interface, registration
Activities/Objects
in the "External World"
Data
Sources
Eye-witnesses (sensors), reports, etc.
Actions


The Information Cycle
"Architecture"
Privacy Firewall
News (Topic) Coverage:
(Topic-"Tracking" Objects:
Encapsulate/update evolving
state of activities or entities in
the external world state)
Archives
Storage Layer
Logical
Representation
(Summarization)
Editors:
(Information
firewall: Filter
coverage)
Decision
Makers:
(Actuators)
Sensor Layer
Sensor interface, registration
Activities/Objects
in the "External World"
Data
Sources
Eye-witnesses (sensors), reports, etc.
Actions


The Information Cycle
"Architecture"
Privacy Firewall
News (Topic) Coverage:
(Topic-"Tracking" Objects:
Encapsulate/update evolving
state of activities or entities in
the external world state)
Archives
Storage Layer
Logical
Representation
(Summarization)
Editors:
Application Layer
Data analysis and Actuation
Sensor Layer
Sensor interface, registration
Activities/Objects
in the "External World"
Data
Sources
Eye-witnesses (sensors), reports, etc.

Where is the Thin Waist?

Where is the Thin Waist?

- Recall:
  - Communication → **Information Distillation**
- A Distillation Socket Layer?

Other Challenges

- Privacy! … did I mention privacy??
- Mobility (opportunistic, symbiotic, parasitic, …)
- Security (control of critical infrastructure?)
- Storage, indexing, archiving, …
- Data analysis, statistical learning, data mining, …
- Design for integration (large number of components breeds integration problems)
- Robustness (adaptation, self-healing, etc)
- Power management
- Performance analysis tools (capacity, availability, …)