The Promise of Sensor Networks to Revolutionize our Environment: Applications and Research

Aris M. Ouksel

The University of Illinois at Chicago

aris@uic.edu

ICPS-06

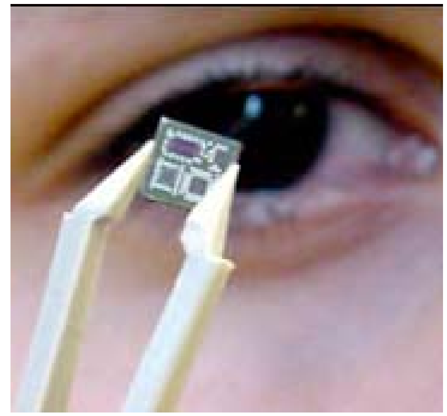
The IEEE International Conference on Pervasive Services

Outline

1. **Disruptive or Incremental Technology**
2. **Vision**
3. **Research Assumptions and Challenges**
4. **Towards a New Theory of Computing**
5. **Rethinking the Protocol Stack**
6. **Cross-Layer Integration and Optimization**
7. **Future Programming Paradigms**

The Embedded Networks Vision

- "Information technology (IT) is on the verge of another revolution… The use of EmNets [embedded networks] throughout society could well dwarf previous milestones."[1]

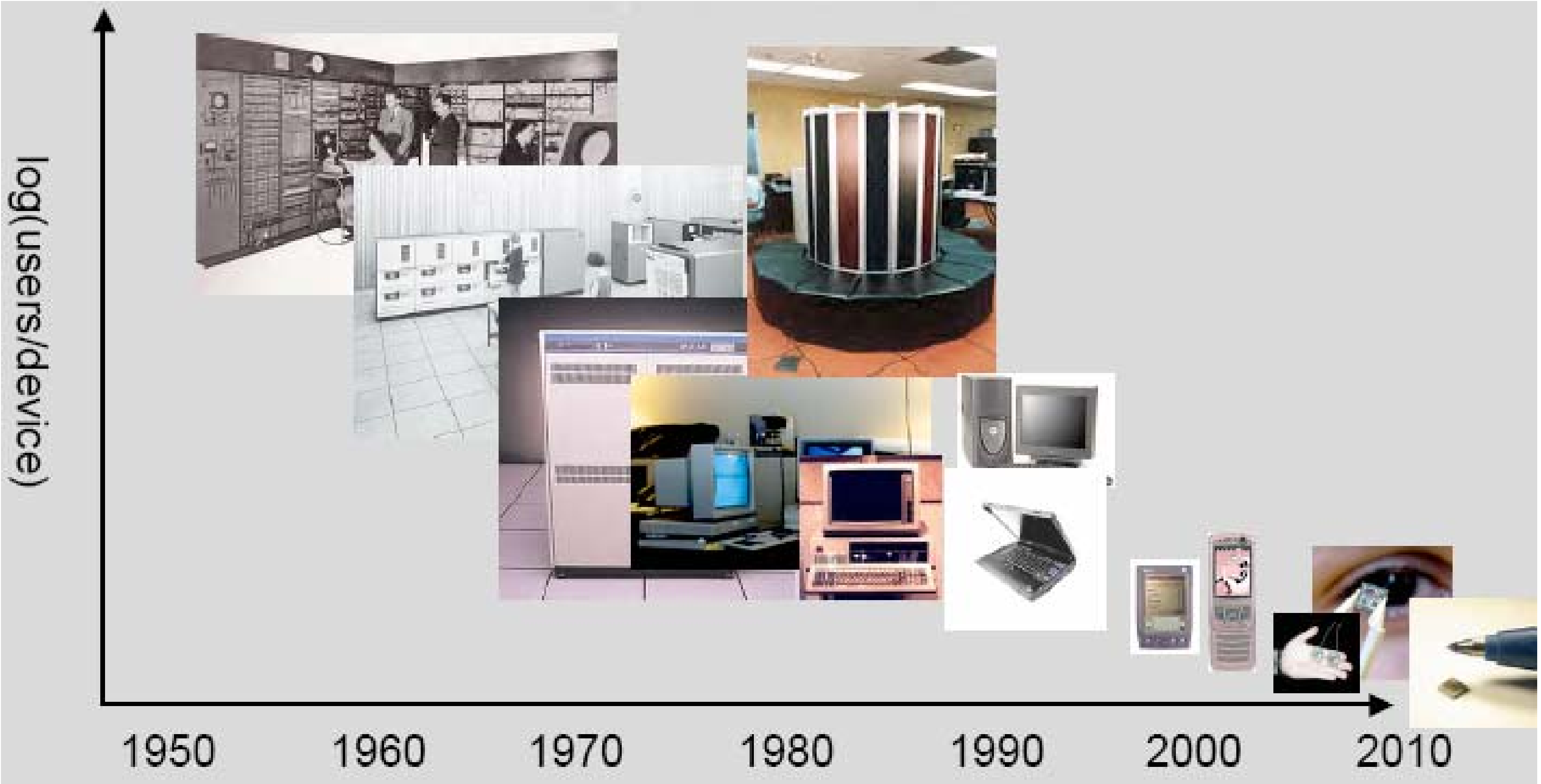


- The motes [EmNet nodes] preview a future pervaded by networks of wireless battery-powered sensors that monitor our environment, our machines, and even *us*."[2]

1 National Research Council. Embedded, Everywhere, 2001.
2 MIT Technology Review. 10 Technologies That Will Change the World, 2003.

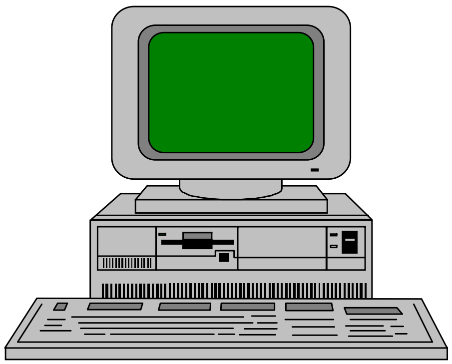
Is Wireless Sensor Technology Disruptive?

Bell's Law

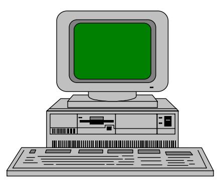


The Trends in Computing Technology

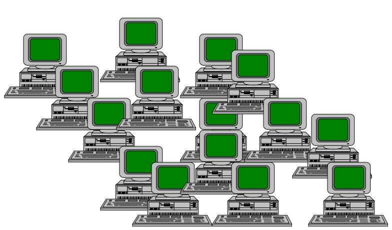
1970s



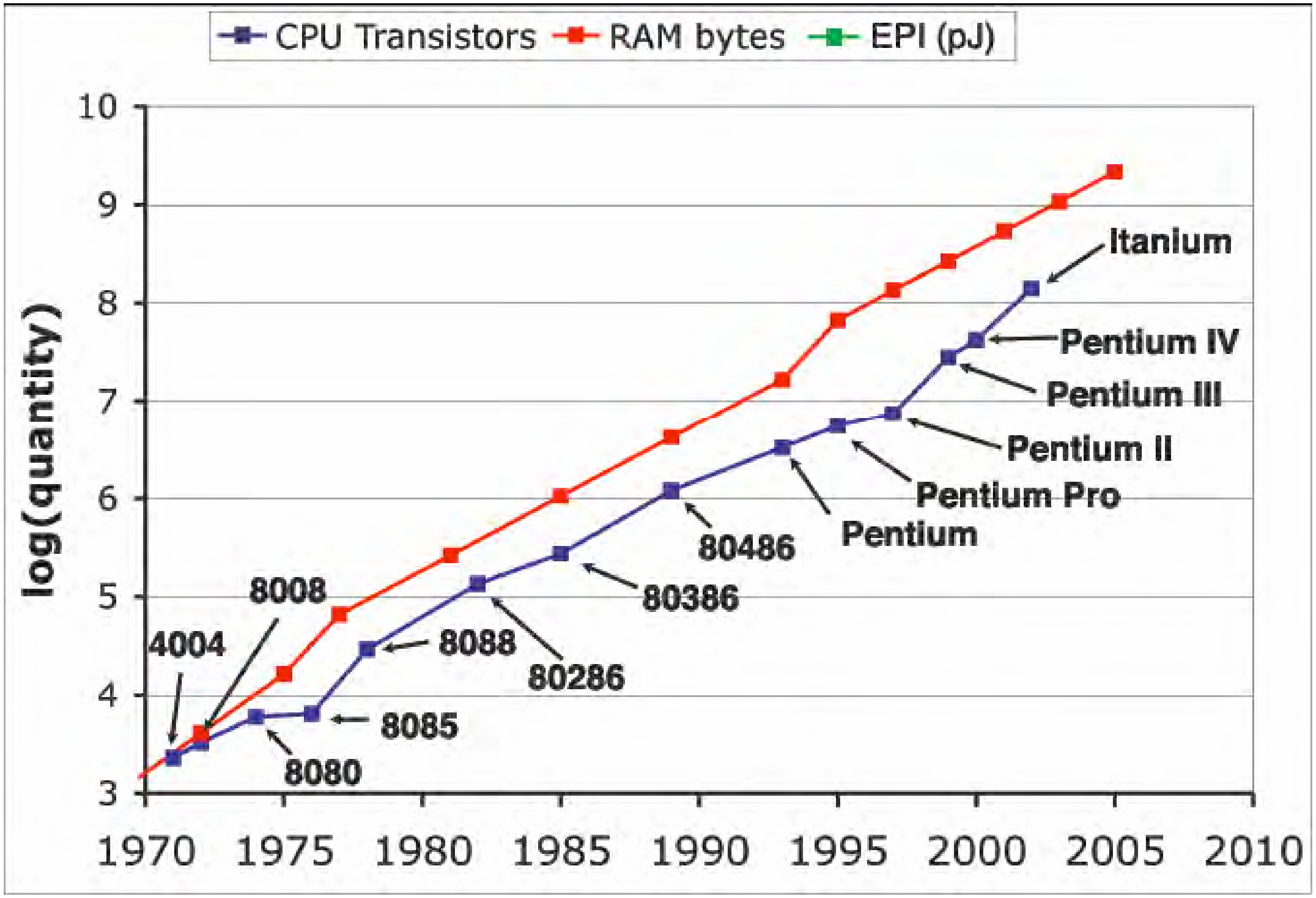
1990s



tomorrow

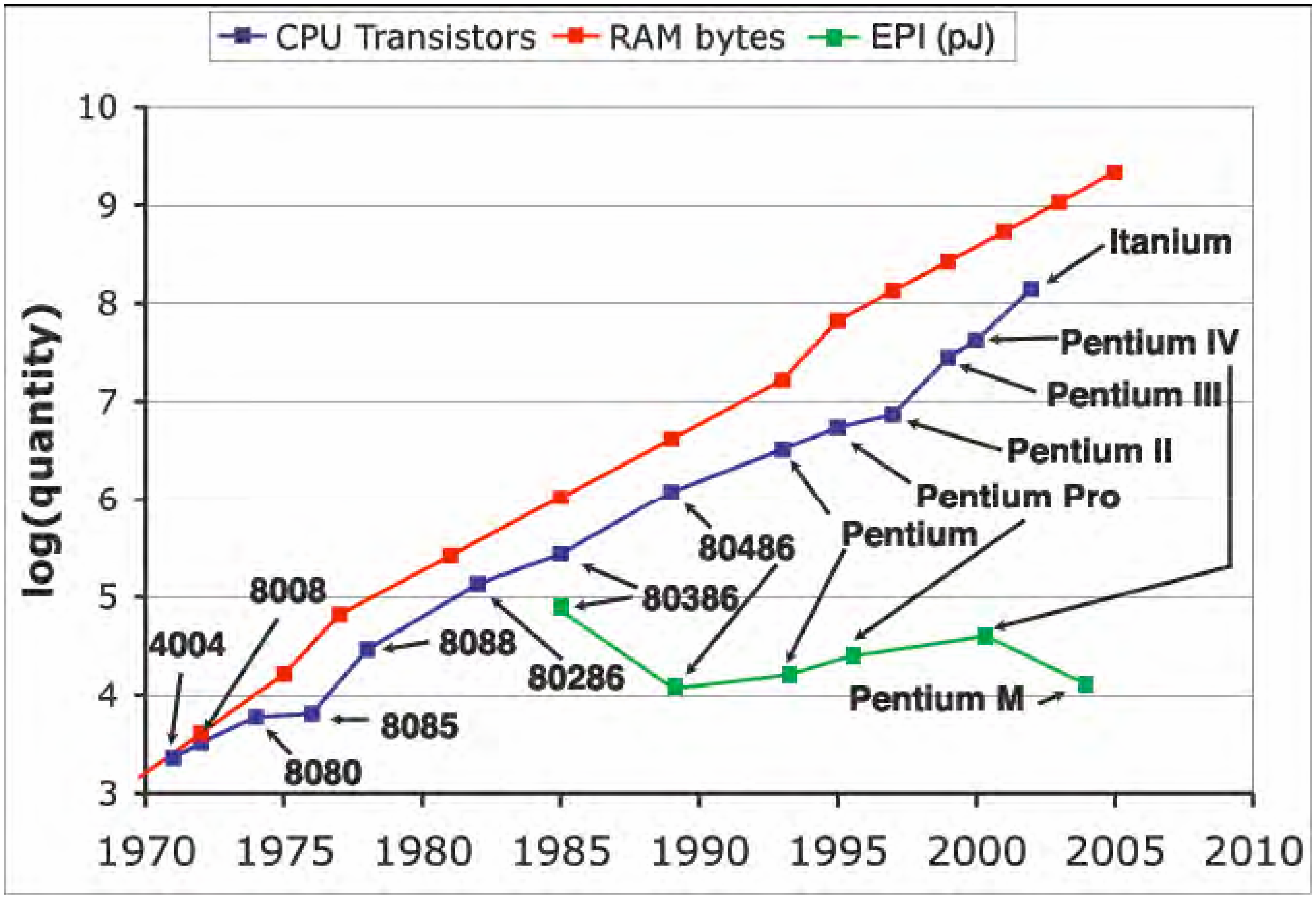


Moore’s Law



CPU Transistors
RAM bytes
EPI (pJ)
log(quantity)
10
9
8
7
6
5
4
3
1970
1975
1980
1985
1990
1995
2000
2005
2010
4004
8008
8080
8085
8088
80286
80386
80486
Pentium
Pentium Pro
Pentium II
Pentium III
Pentium IV
Itanium

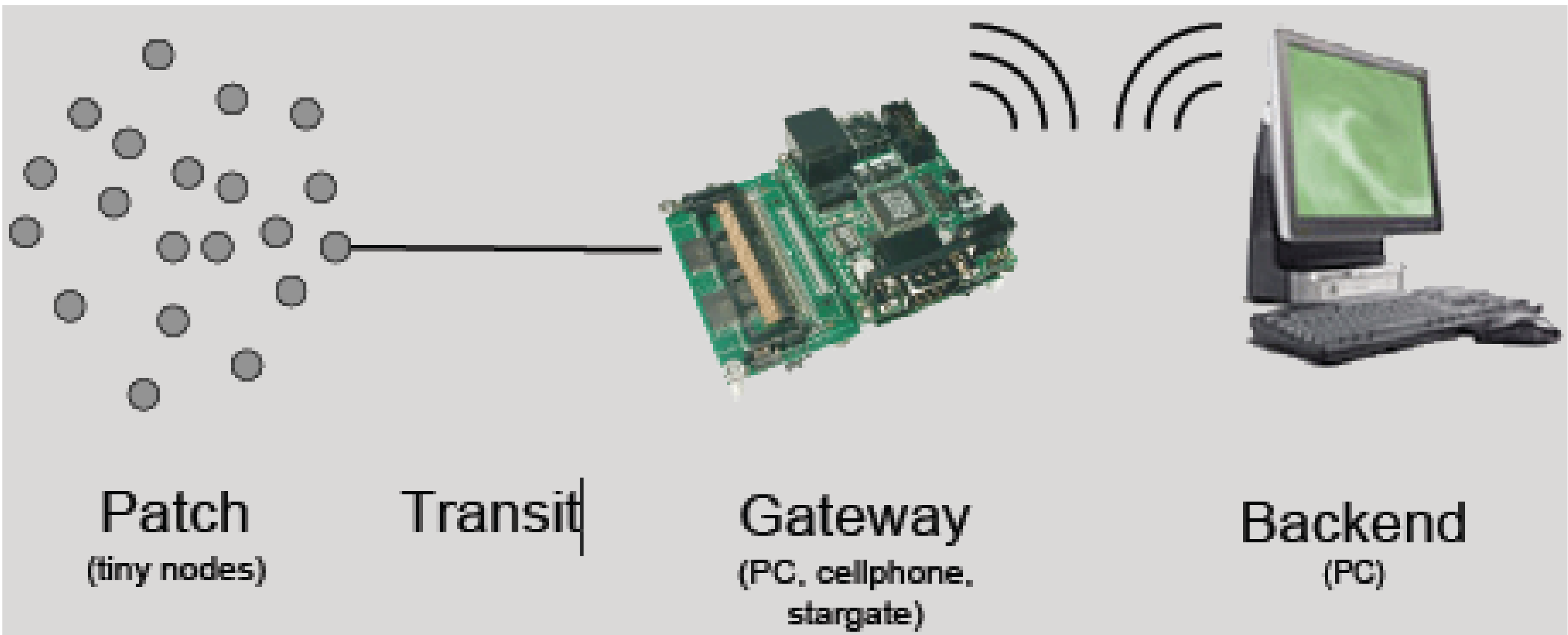
Moore's Law with Energy



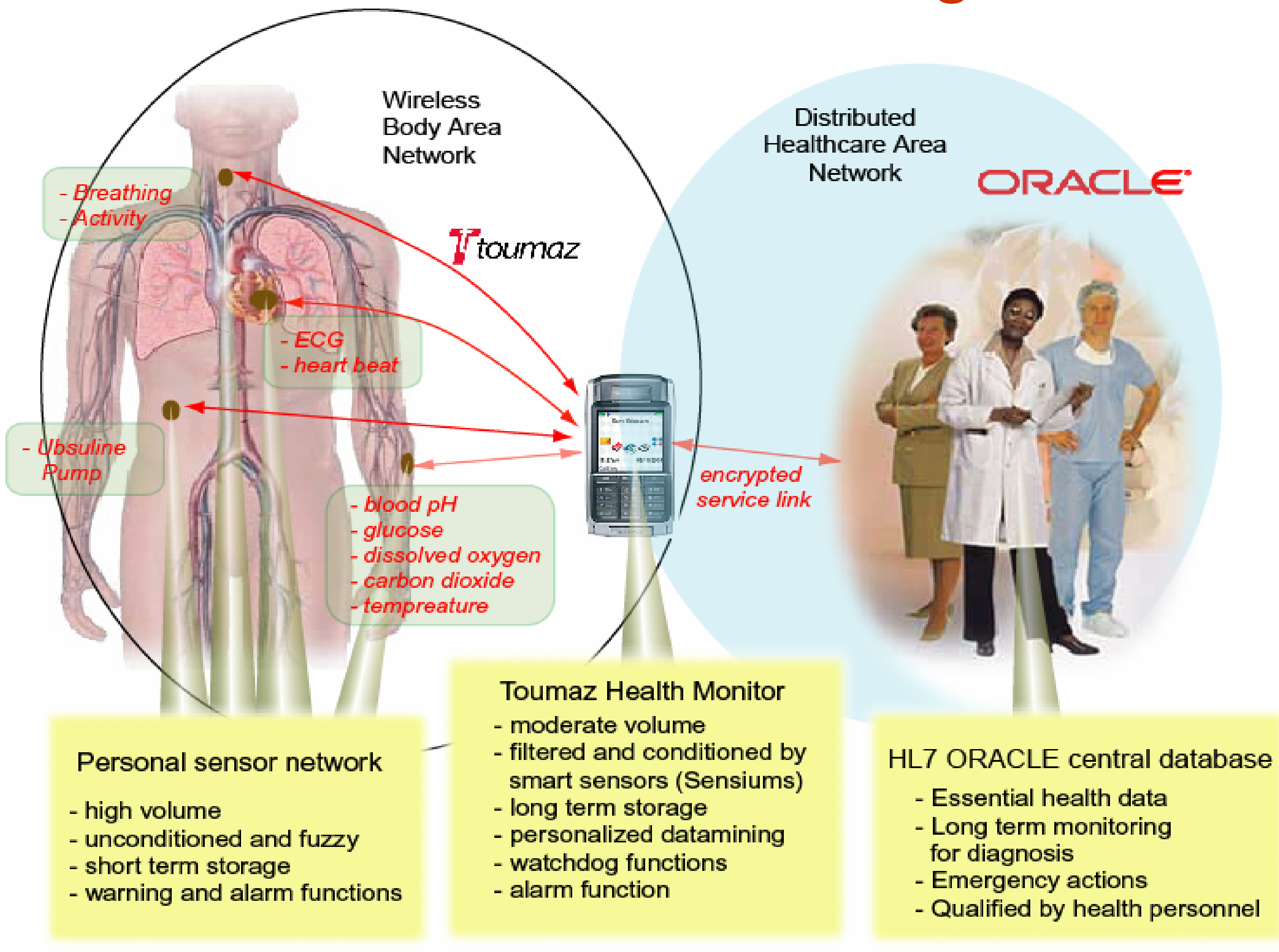
Many Tiny Low-Cost Devices

- Weighing the costs
  - Cost of device
  - Cost of deployment
  - Cost of maintenance
- Unseen and in uncontrolled environments
  - A tree, a body, a faucet, a river, a vineyard
- Wireless is inherent to embedded sensor networks
  - Reduces cost of deployment and maintenance
  - Wires not feasible in many environments
- Mobility

Sensornets Today



Management of Chronic Disease
Pro-active Monitoring

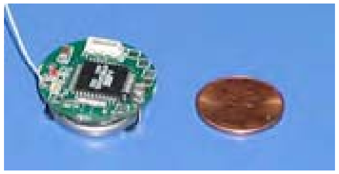


The Hardware

- Two platform classes: gateway and embedded wireless

Linux: MB of RAM
Active power: W
Sleep power: mW

3 orders of magnitude



TinyOS: KB of RAM
Active power: mW
Sleep power: µW

- Energy is defining metric: lifetime, form factor, resources

AA Battery for a year: ~2.7 Ah / (365 days * 24 hours) 300µA avg. draw

TakeAways

- Cost, scale, lifetime and environment require wireless
  - Wireless makes energy the limiting factor
  - Moore’s Law has not followed an energy curve
  - Need for long-lived deployments means that ultra low-power nodes must still spend 99% of their time asleep

Rethinking the Fundamentals

Extreme energy limitations, coupled with long lifetimes, large numbers, and embedment, completely change hardware design, software design, OS structure, network protocols, and application semantics.

Hardware
Miniaturization

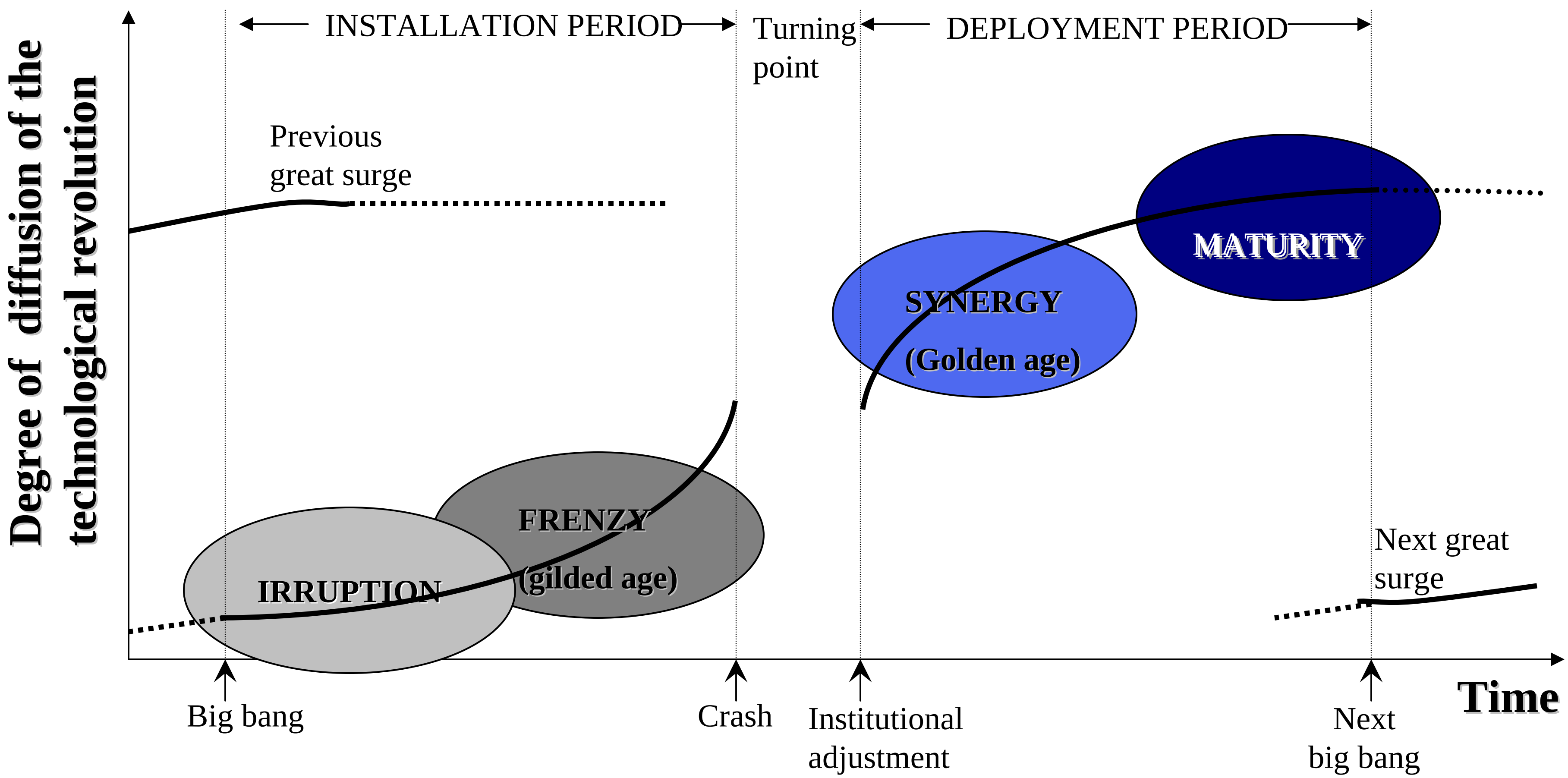
Software
New programming Abstractions

Communication
New Protocol Stacks

Theory
Scalable emergent Behavior

The life and times of a technology

Recurring phases of each great surge

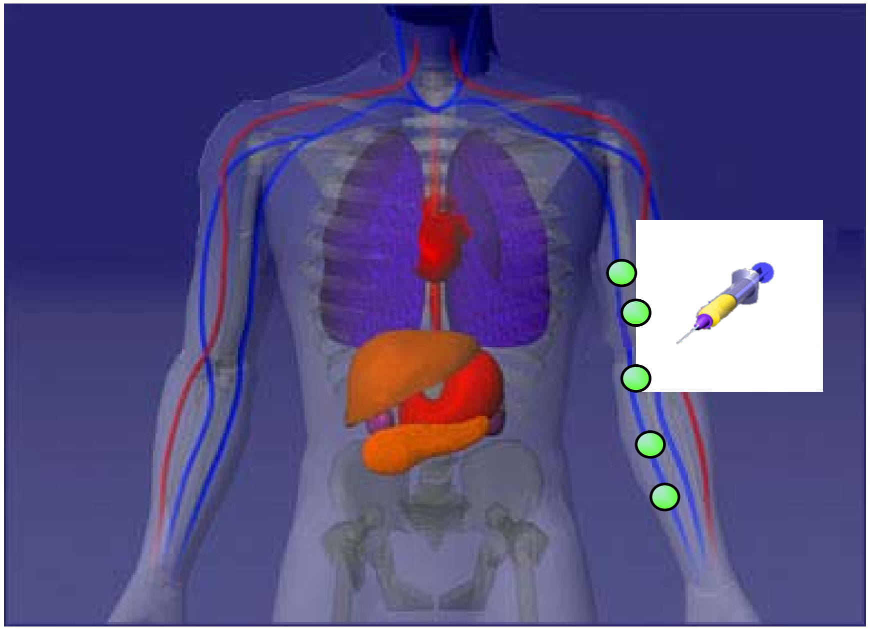


Source: Carlota Perez

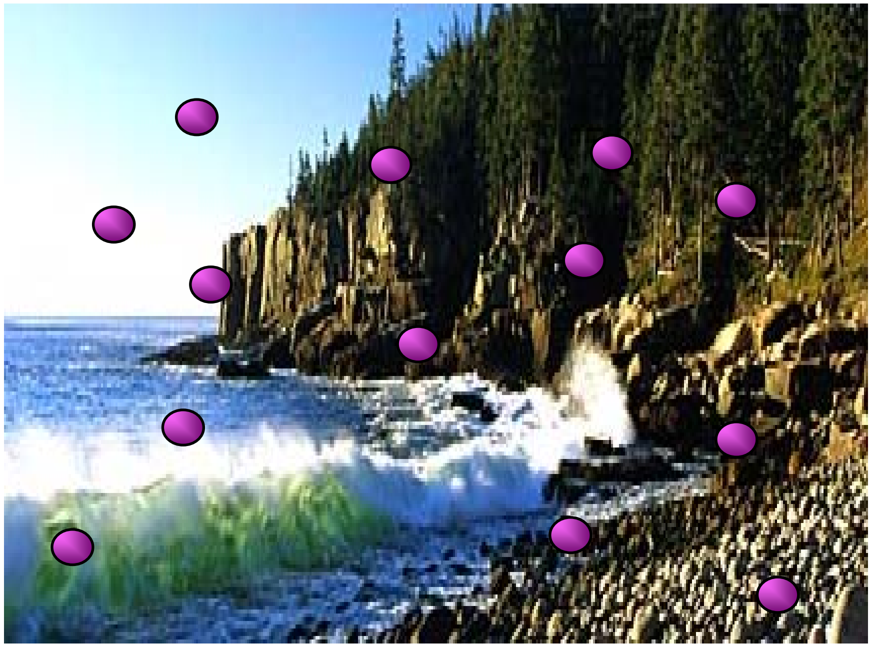
VISION

Vision

- **Embed** numerous sensors in the physical world: monitor and interact.
- **Gather** temporal and spatial information from sensors.
- **Enable** self-organization/ coordination capabilities in large network of sensors for high-level tasks.
- **Achieve** robust distributed systems.



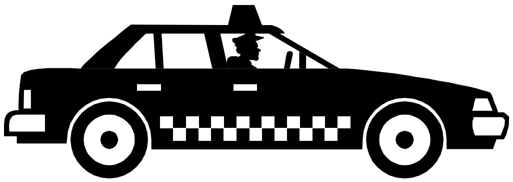
UCLA, LECS



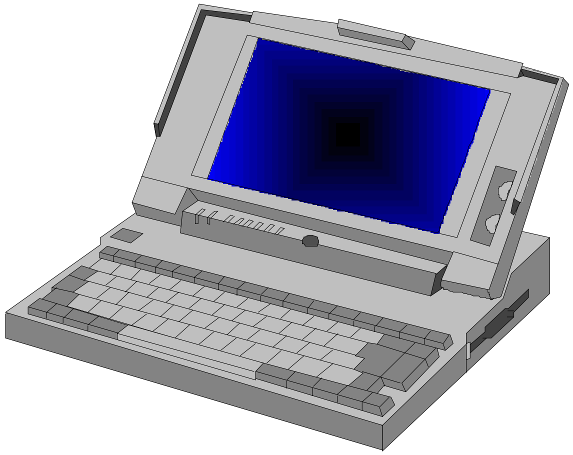
http://cougar.cs.cornell.edu

Embedded Systems

- Engine control
- Wristwatch
- Modems
- Mobile phone
- Internet appliances
- Process Control
- Air Traffic Control
- 60 Processors in Limo

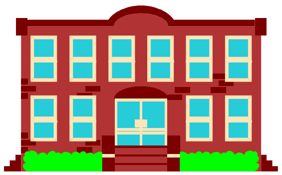


- Smart Spaces
- Sensor/Actuator/CPU clouds with movable entities
- Smart dust



Smart Spaces

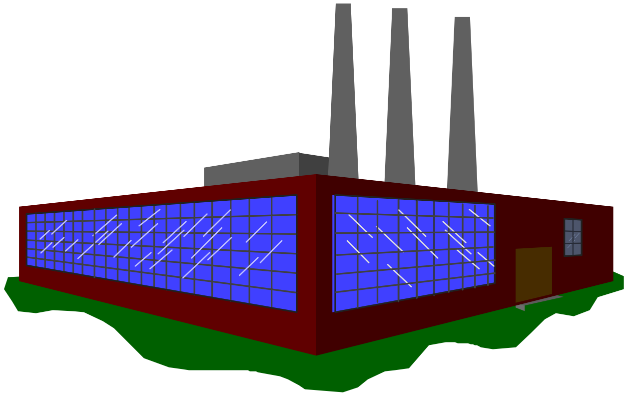
- Pervasive
- Global Connectivity



Smart School

Smart Classroom

Smart City

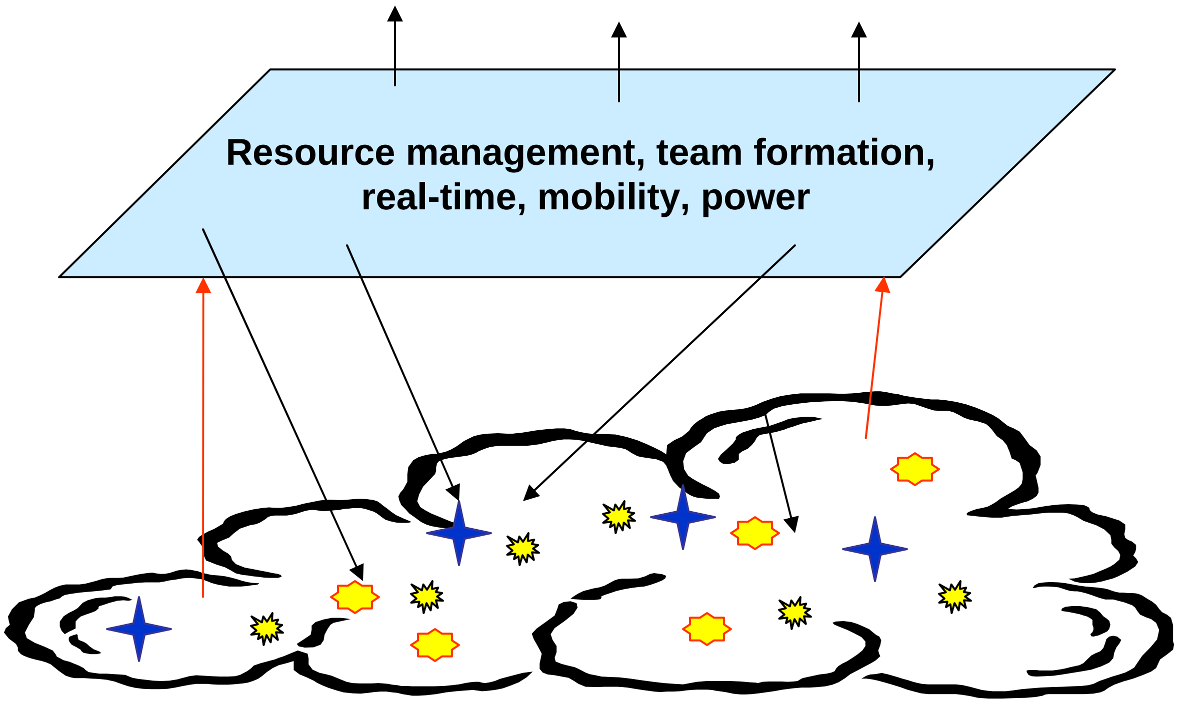


Smart Factory

Applications

- Counter-Terrorism
- Personal Security
- Habitat Monitoring
- Traffic Surveillance and Control
- Emergency Scenarios

Sensor/Actuator Clouds



Heterogeneous Sensors/Actuators/CPUs (or Homogeneous but Powerful)

Smart Dust – Biological metaphor

- battlefield awareness
- earthquake response
- tracking movements of animals
- smart paint
- MEMS in human bloodstream

Key Issues

- Enormous numbers of devices and amounts of software needed
  – flexible and tailor-able
  – interaction with physical/distributed environment (of greater heterogeneity - not just cpus)
- Aggregation - system as a whole must meet requirements
  – individual entities not critical
- *Real-Time, Power, Mobility, Wireless, Size,* Cost, Security and Privacy

Hardware Technology

Towards nano-scale devices

Sensor node platforms
(hardware)

Node	Bandwidth	MIPS	Flash	RAM	Duty Cycle %
Spec	50k	5	0.1	3k - 4k	0.1-0.5
Mote	75k - 250k	10	<0.5 m	10k	1-2
Imote	500k (Bluetooth-based)	50	<10m	128k	5-10
Stargate	10m	400	32m	512k - 64m	50-100

Based on J. Hill, M. Horton and R. Kling (ACM Comm. June 2004) Updated February 2006

Scaling Dense WSNs

Year	Nodes	Area	Lifetime	Program Size
2000-2	~10	~10 sq.m	5 days	~5KB
2003	~100	~10^3 sq.m	5days-1year	~30-100KB
2004-5	~1000	~10^6 sq.m	1month-1year	~200-500KB

Increase in:

- Component depth and interaction complexity
- Component unreliability and variability
- Deployment and manageability

Characteristics of Sensors and Sensor Networks

- Sensing Capabilities:

  Chemicals, radiation levels, light, seismic activity, motion, audio, video
- Unattended and Untethered “control systems”
- Technology Challenges:
  - Battery lifetime and Energy Consumption
  - Miniaturization
  - Low Bandwidth
  - Low computation capability

The Hardware Challenge

- Miniature hardware devices must be manufactured economically in large numbers
- Current microprocessor manufacturing technology will soon reach its lithographic size limits
- What are possible alternative future technologies?

Cellular Computing

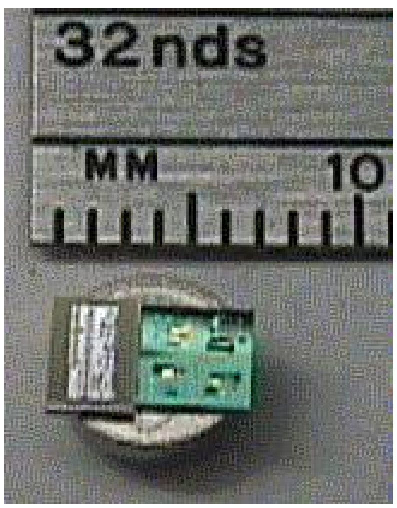
- Cells as logic gates
- Basic inverter: Concentration of protein Z is inversely proportional to concentration of protein A.
- NAND gate: Production of protein Z is inhibited by presence of proteins A and B.

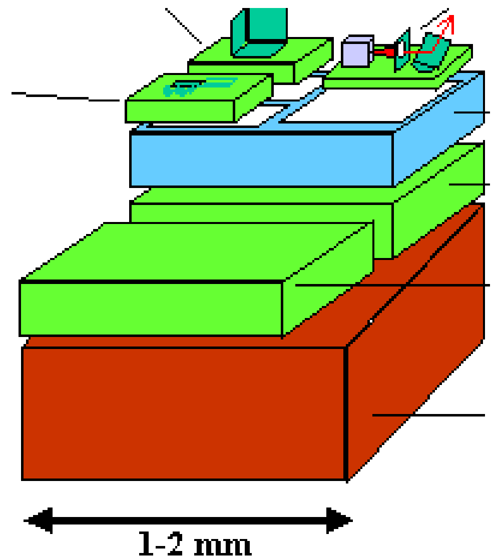
Nano-scale Computing

- DNA manipulation can organize cells into precisely engineered patterns
- This technology could be the foundation for construction of complex sub-nano-scale extra-cellular circuits:
  - Biological system – machine shop
  - Proteins – machine tools
  - DNA – control tapes
- Circuits are fabricated in large numbers by cheap biological processes

Smart Dust

- Current technology: 5mm motes
- Goal: 1mm





Research Assumptions and Challenges

How the Problems Change

- Environment
  - connect to physical environment (large nos., dense)
  - massively parallel interfaces
  - faulty, highly dynamic, non-deterministic
  - wireless
- Network
  - structure is dynamically changing
  - sporadic connectivity
  - new resources entering/leaving
  - large amounts of redundancy
  - self-configure/re-configure
  - individual nodes are unimportant

How the Problems Change

- OS/Middleware
  - manage aggregate performance
    - *control* the system to achieve required emerging behavior
  - move nodes to area of interest (self-organizing)
  - fuzzy membership and team formation
  - manage power/mobility/real-time/security tradeoffs
  - geographically based (data centric)

Implications

- Fundamental Assumptions underlying distributed systems technology has changed
  - wired => wireless (limited range, high error rates)
  - unlimited power => minimize power
  - Non-real-time => real-time
  - fixed set of resources => resources being added/deleted
  - each node important => *aggregate performance*
  - ...
- New solutions necessary

Implications

- What a single node knows is less important
  - iterative, diffusion, and masking type algorithms
  - neural net?
  - Adaptive control with compensation

- Resource Management
  - too many communication errors (feedback control) => move closer, increase power ...

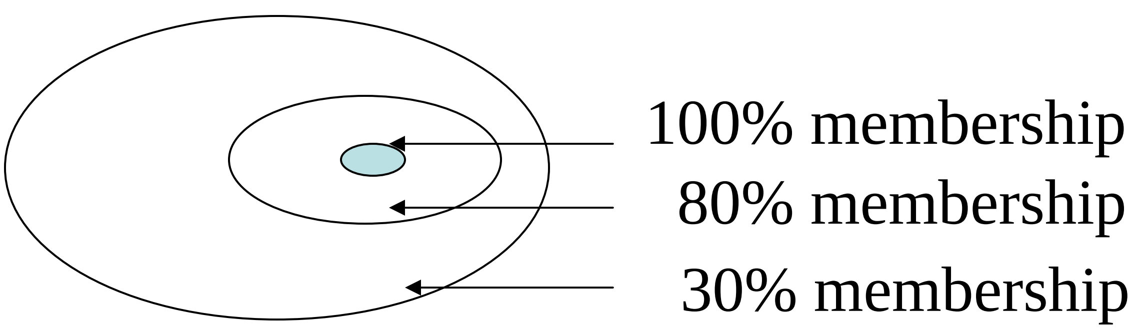
Example: Consensus

- Classical consensus: all correct processes agree on one value
  - No power constraints
  - No real-time constraints
  - Does not scale well to dense networks
  - Approximate agreement (some work here) - on sets of values (physical quantities)

- Solutions
  - diffusion and aggregation
  - Density/topological maps

Example continued

- 1000 nodes to produce signal strength above a threshold
  - 500 enough
  - turn off others to save power
  - Don’t want to know which nodes have failed; individual nodes not important
- Topological model



Aggregate Performance

- Specify and control emerging behavior to meet system-level requirements
  - Smart Spaces
  - Smart Clouds of sensors/actuators/cpus
  - Smart Dust

Towards a New Theory of Computing

Local algorithms, scale and emergent behavior

Aggregation Behavior
Self-Organization
Activity-Driven Deployment

Aggregate Behavior Theory

- Current distributed algorithms typically describe interactions of a finite number of powerful machines
  - Such distributed algorithms have scalability problems
- How to develop computation models for an “infinite” number of simple devices?
- Can we develop algorithms to perform better with *increased* scale?

Analogy

- In many physical and biological systems noisy local component interactions generate robust aggregate behavior *by virtue of scale*
  - Emergence of bulk properties of matter from local atomic interactions
  - Formation of complex biological organisms from local cellular interactions
  - Phase transitions resulting from local molecular interactions

Main Challenge

- How does one engineer pre-specified behavior from cooperation of immense numbers of unreliable parts linked in unknown irregular ways?
  - New approaches to fault-tolerance and aggregate behavior

- How to design local interactions to produce an aggregate behavior of choice?

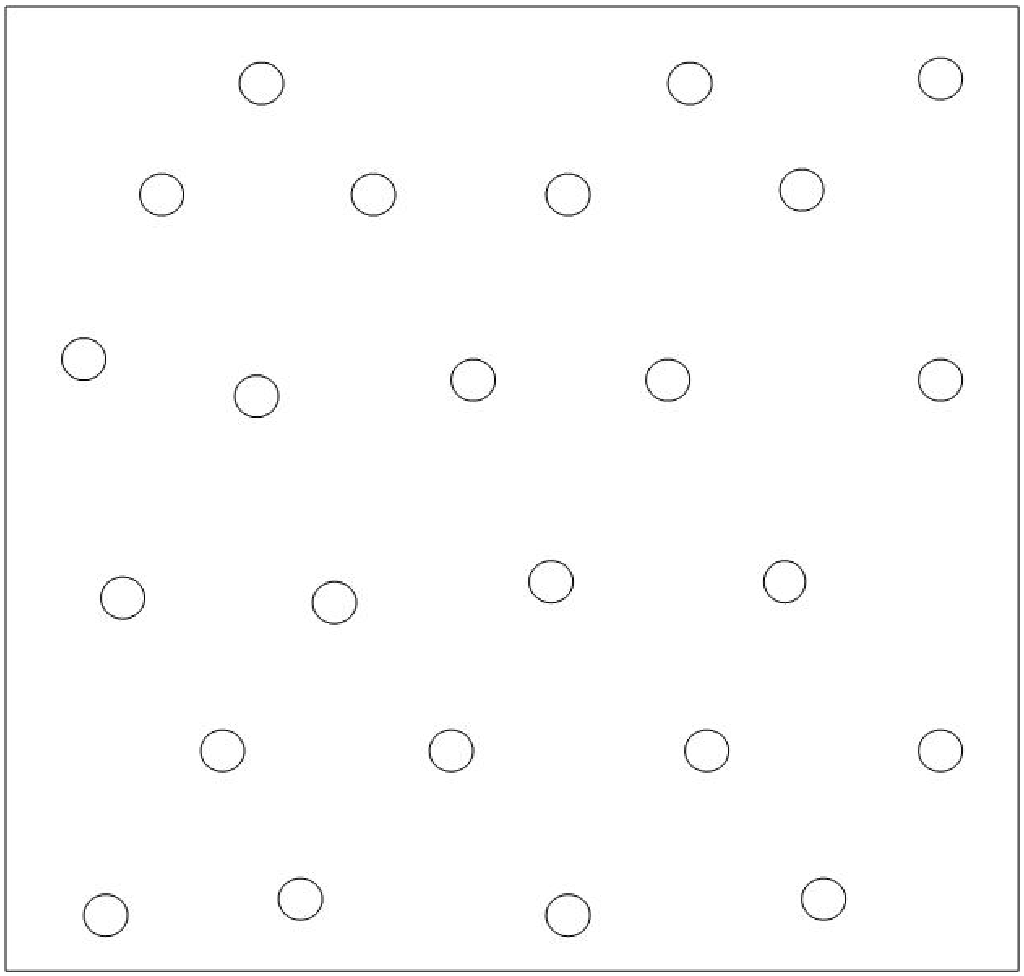
Scalable Coordination and Self-Organization

- Algorithms are needed to self-organize large numbers of nodes
  - Clustering algorithms
  - Team formation
  - Approximate consensus
  - Triangulation
  - Routing and communication

Research Challenges

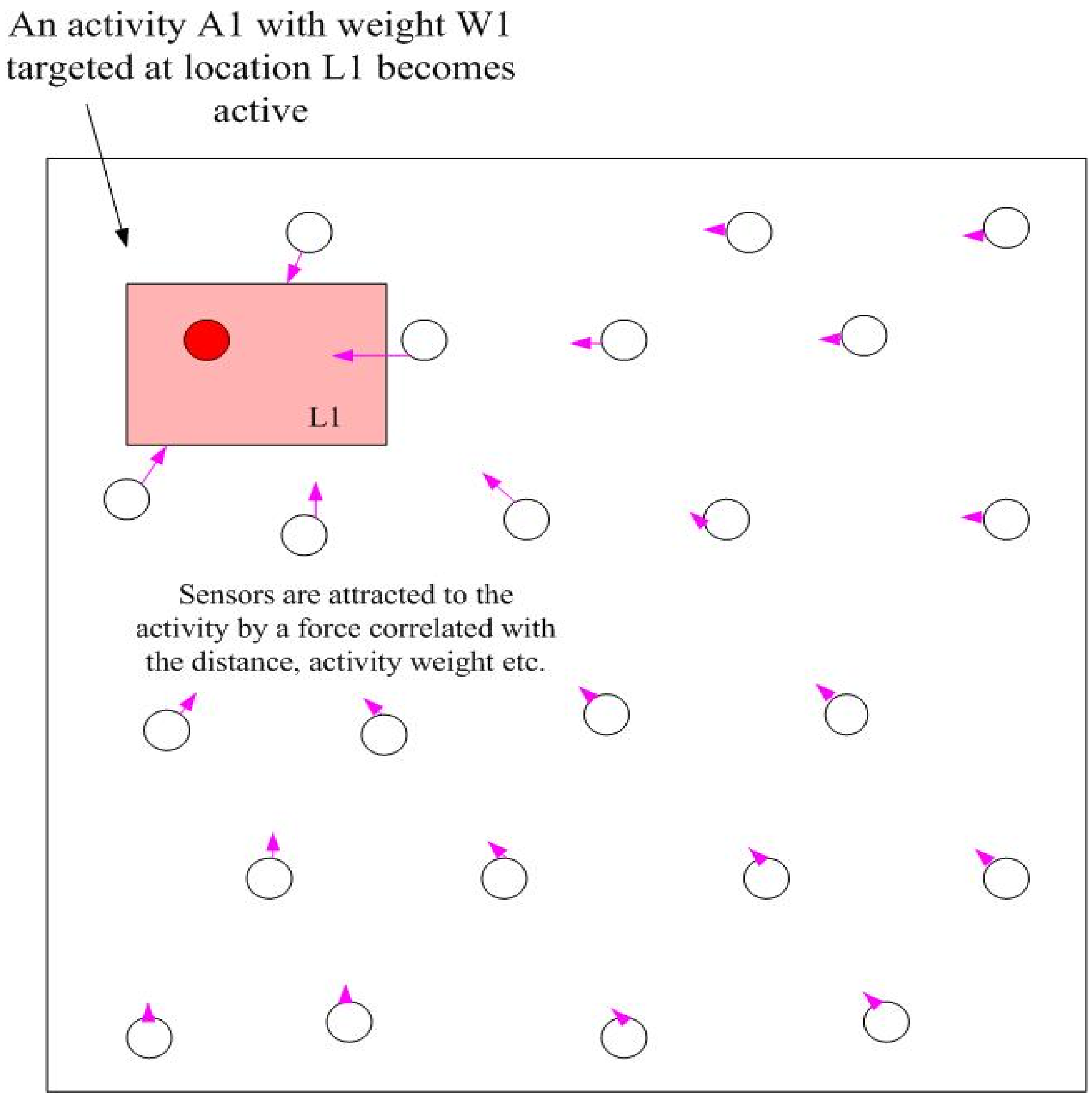
- **Self-Organizing and Self configuring systems that can be deployed ad hoc**
- **New information processing techniques:**
  - Measure and adapt to unpredictable environment
  - Heterogeneity of sensors and their capabilities
  - Spatial diversity and density of sensors
  - Aggregation and Approximation
  - Streaming data
- **Mobility**
- **Privacy and Security**

Initial sensors distribution – no activity

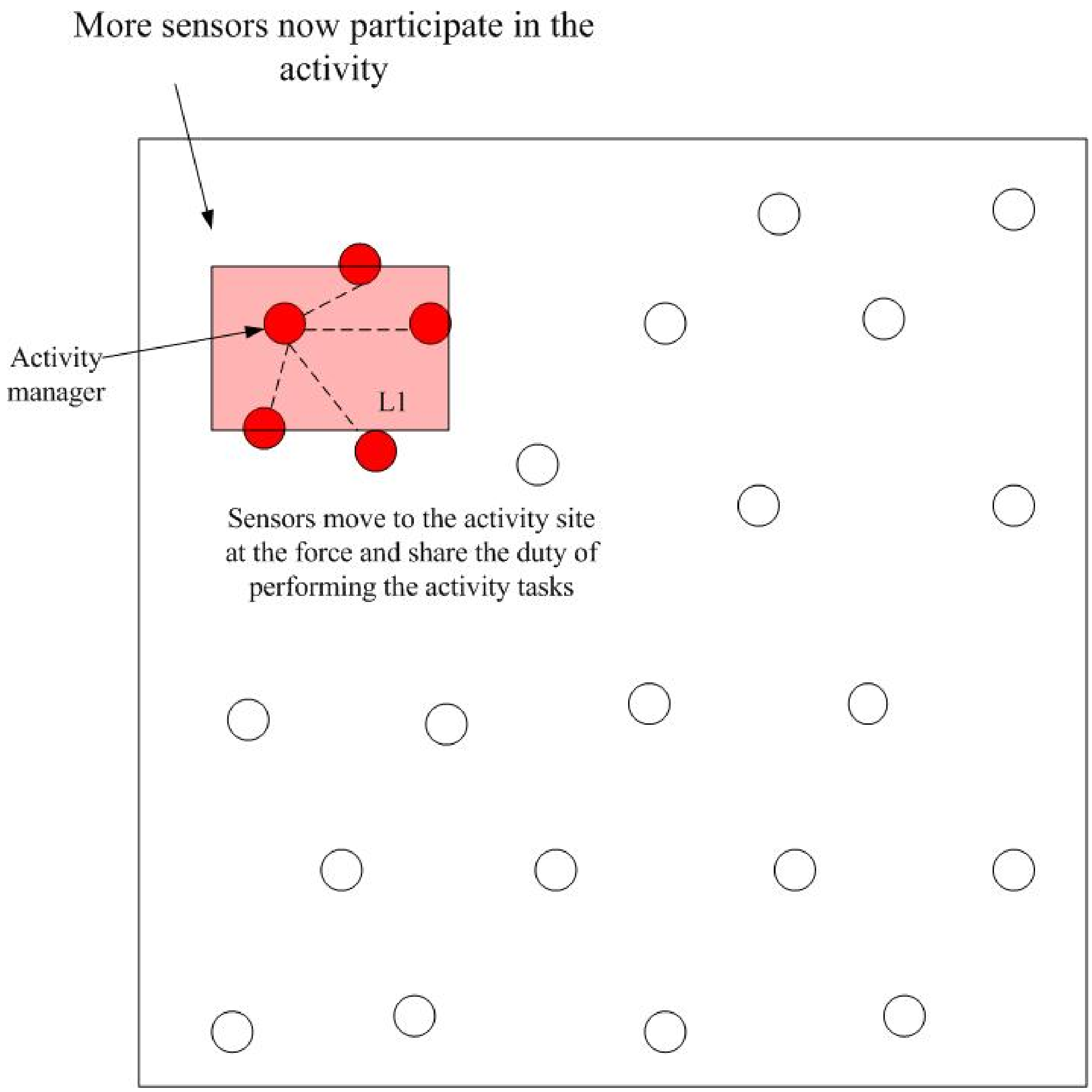


An activity becomes alive …

Forthcoming: Aris M. Ouksel and Lin Xiao, “Activity-Driven Indexing: A Dynamic System Approach”, 2006.

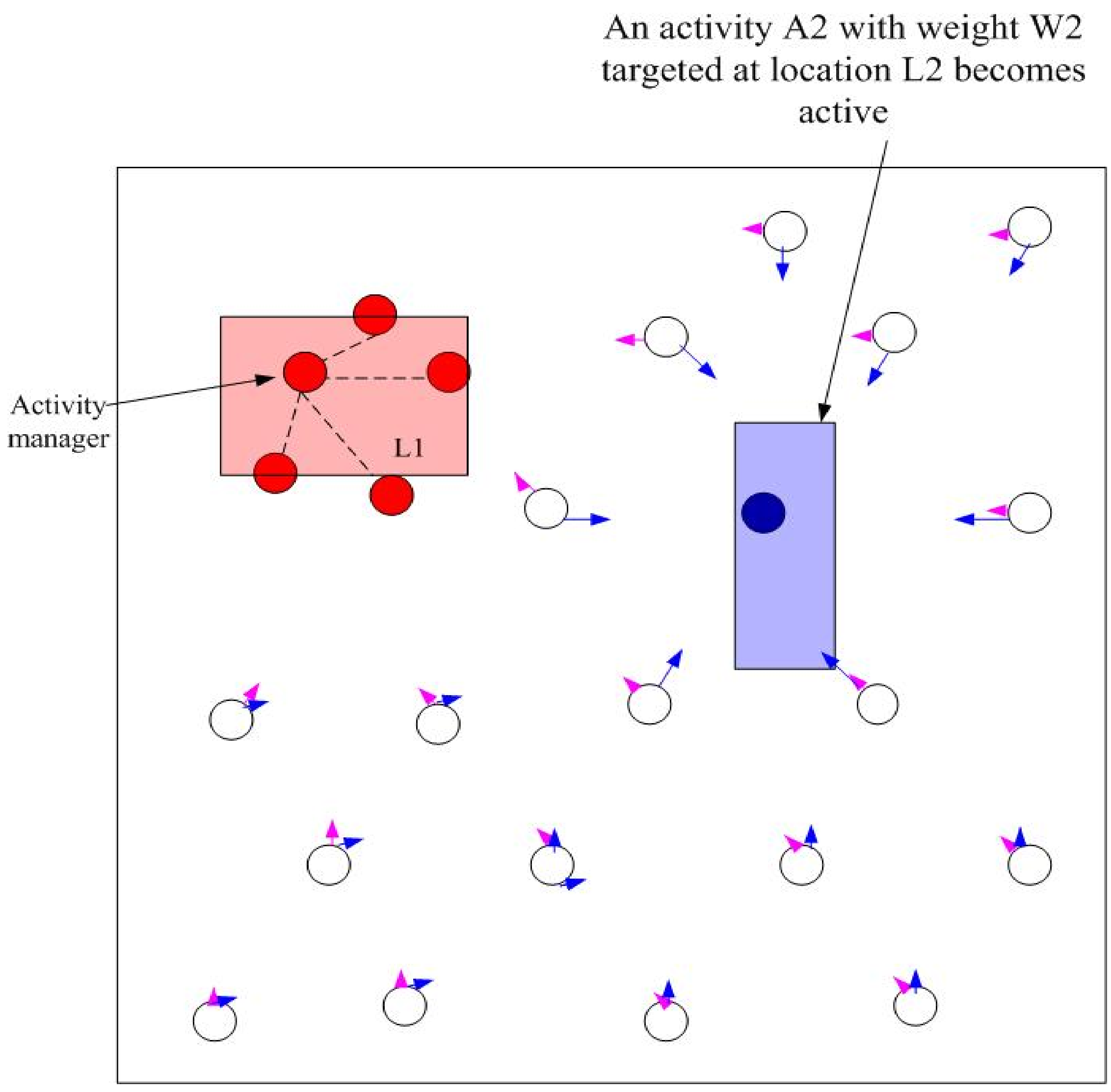


As a result …



More sensors now participate in the activity
Activity manager
L1
Sensors move to the activity site at the force and share the duty of performing the activity tasks

Another activity becomes alive



Conclusions

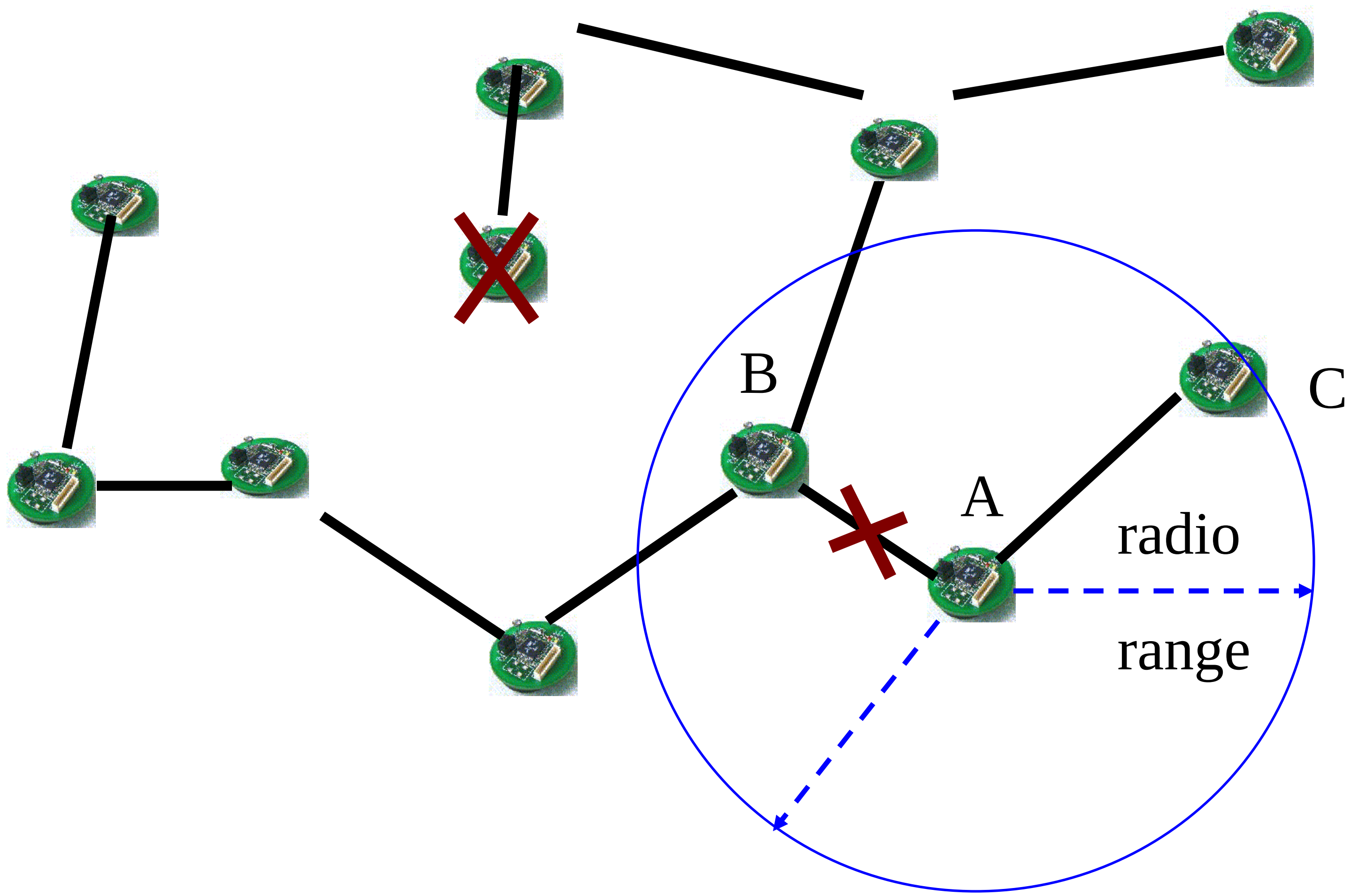
- **Data processing inside the network**

- **Adaptive localized algorithms to achieve desired global behavior**
  - Dynamic environments preclude pre-configuration
  - Centralization to dynamic state information is unaffordable

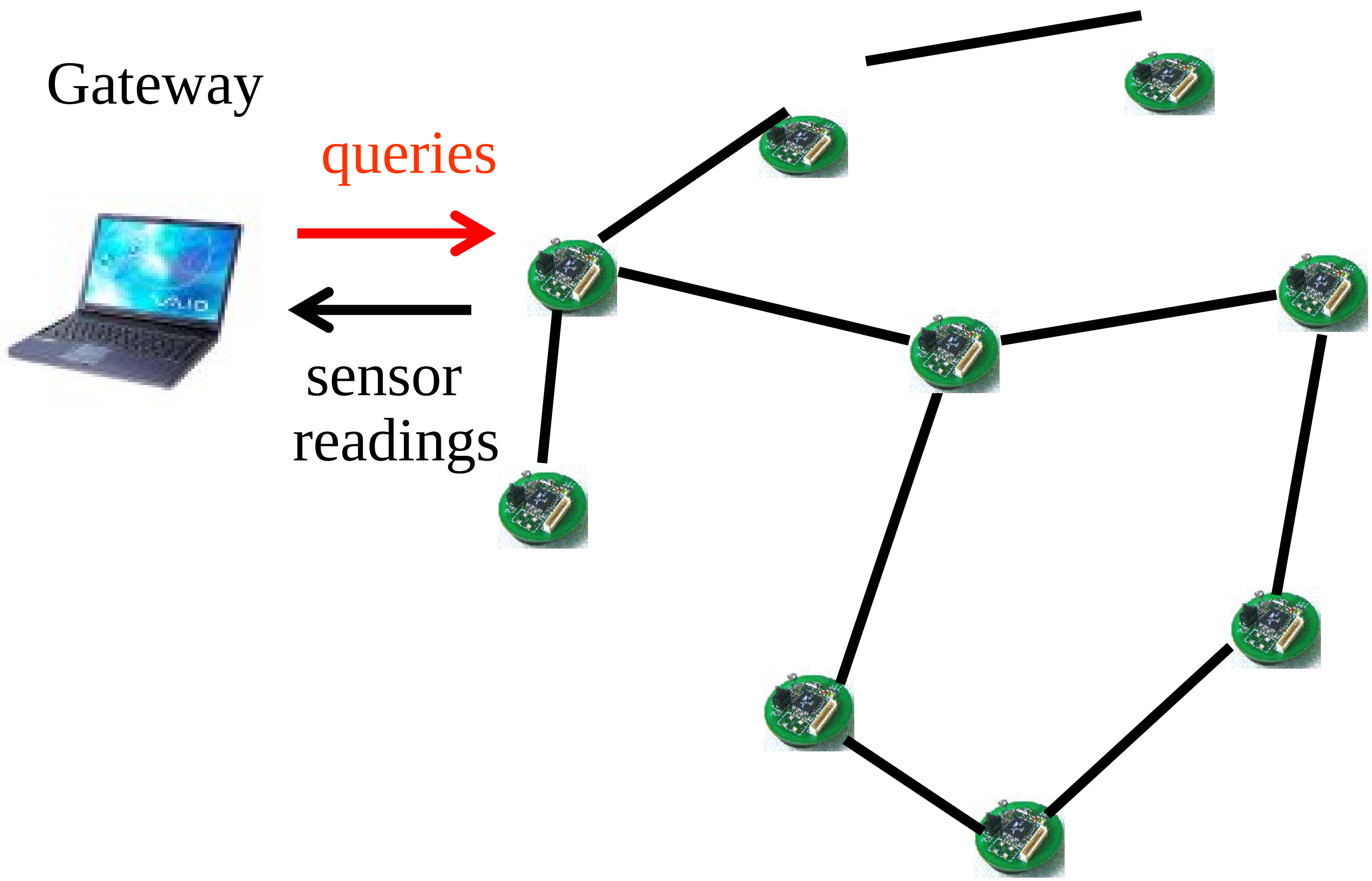
Deployment Architectures

Sensor network

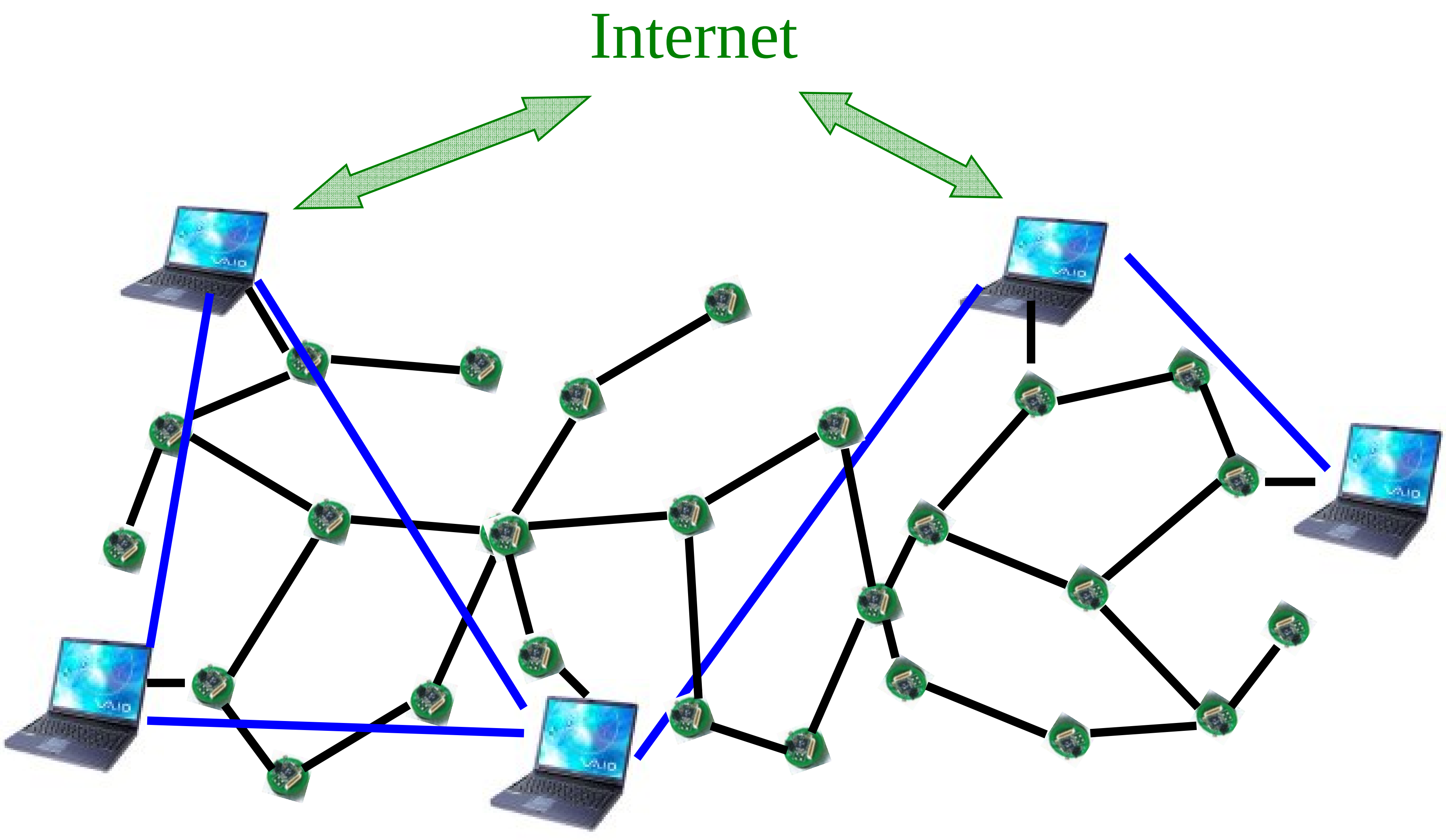
- A collection of sensor nodes deployed in an area and connected through a multi-hop wireless network.



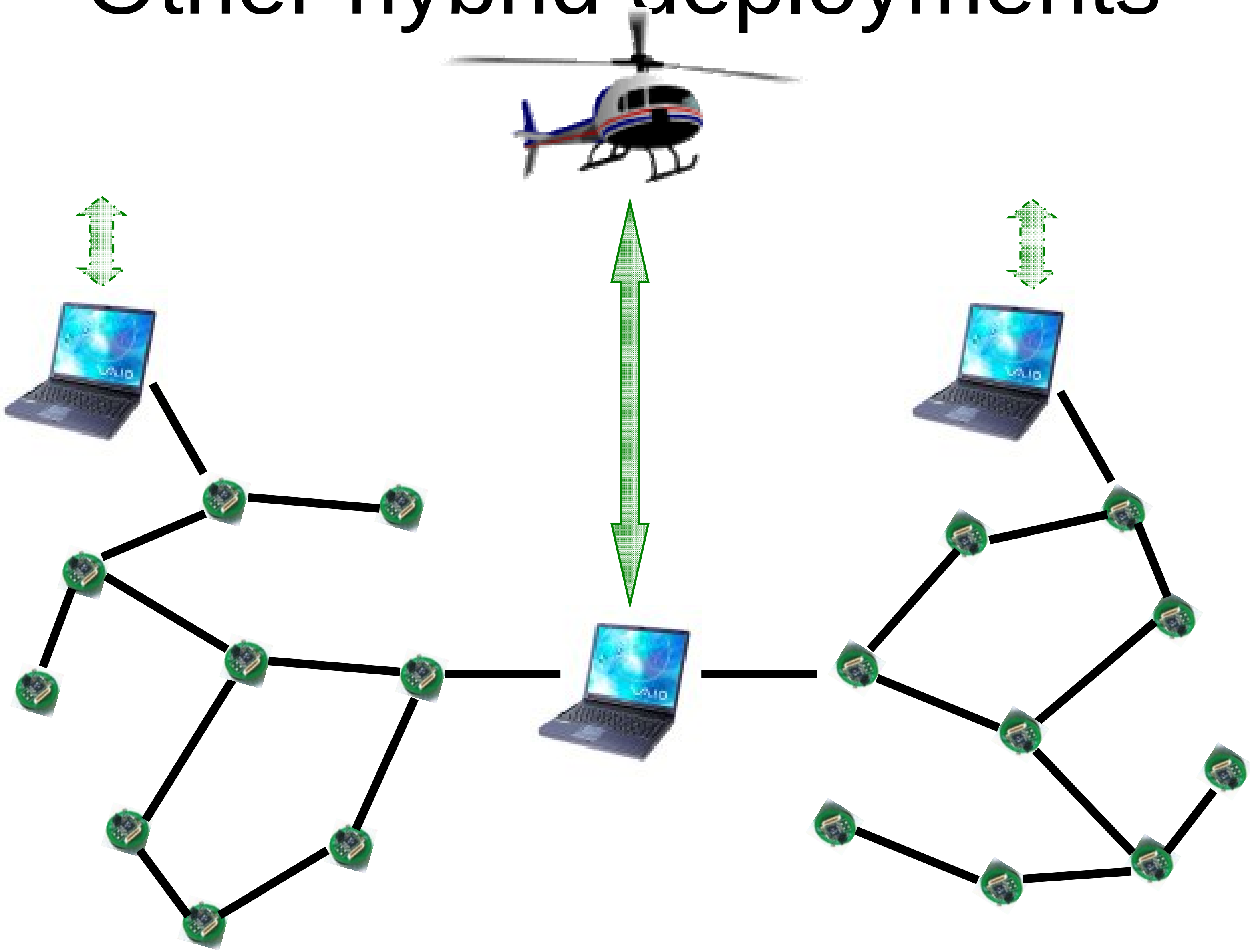
Simple deployment



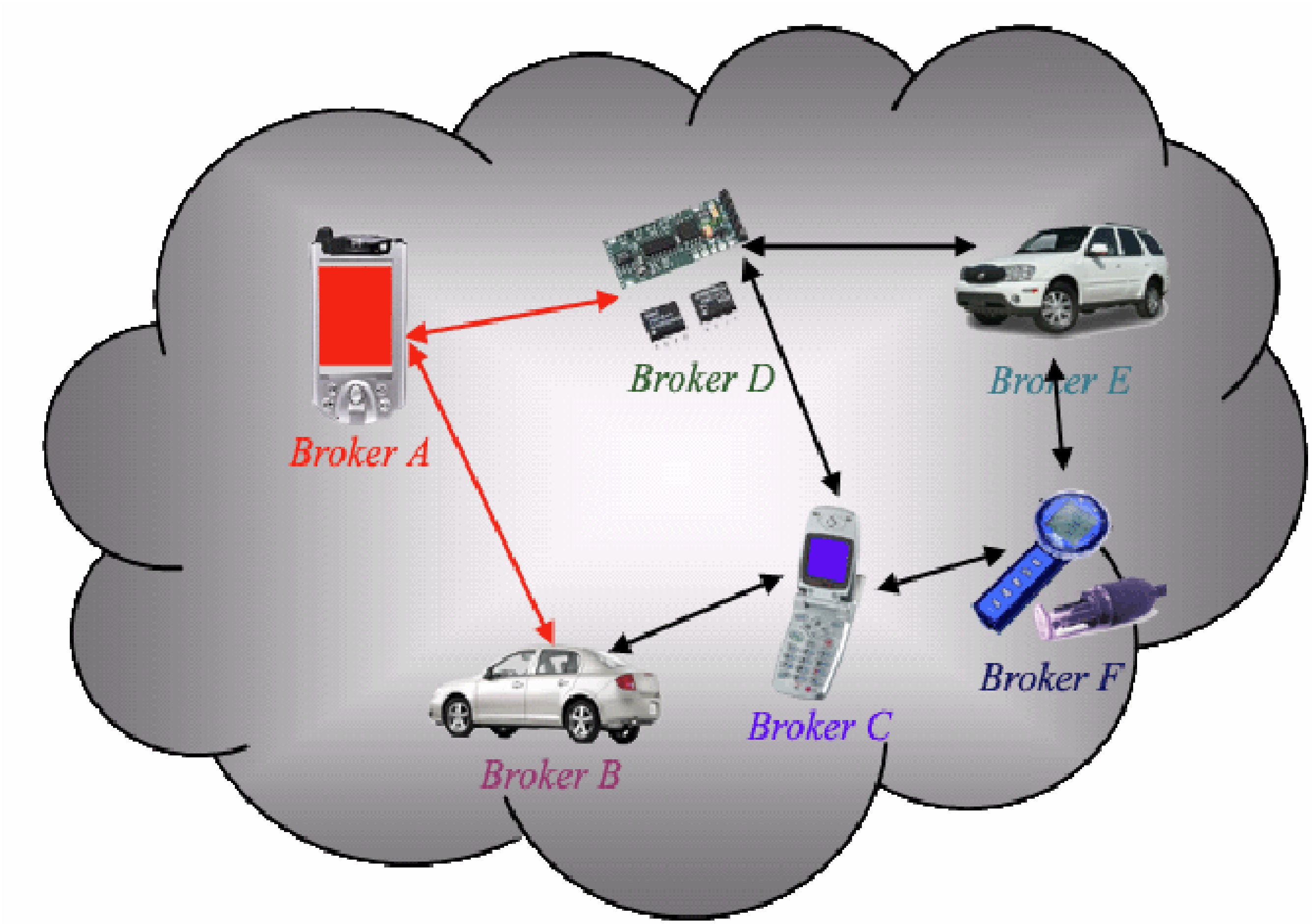
Hierarchical deployment



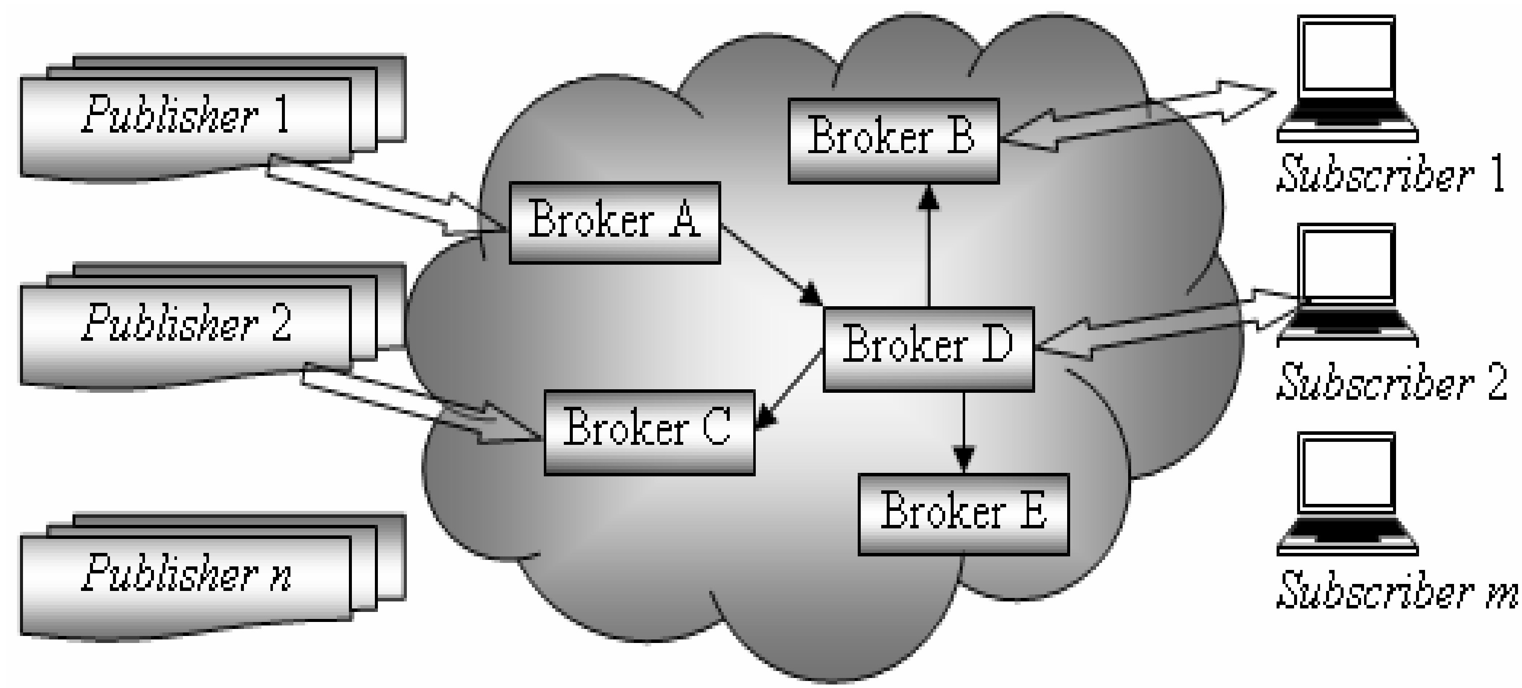
Other hybrid deployments



Publish/Subscribe in MANET



Architecture: Pub/Sub Paradigm



Rethinking the Protocol Stack

Towards new communication paradigms

The Communication Protocol Stack

Internet	Sensor Net
Transport (TCP, UDP)	Transport (Area-to-Area)
Network (IP)	Network (Diffusion; ID-less nodes)
Link (Ethernet,...)	Link (Low Power Wireless)

Low Power Communication

- Radio communication
  - Consumes too much power
  - Requires a large antenna
- Optical communication
  - Base-band communication
  - No modulation, active filtering, demodulation
  - Beam can be aimed at destination

Challenges

- Line of sight requirement
- Aiming requirements
- Matching the field of view
- Link directionality
- Bit-rate, distance, energy tradeoffs (for given SNR)
- Multi-hop optical routing
- Mobility

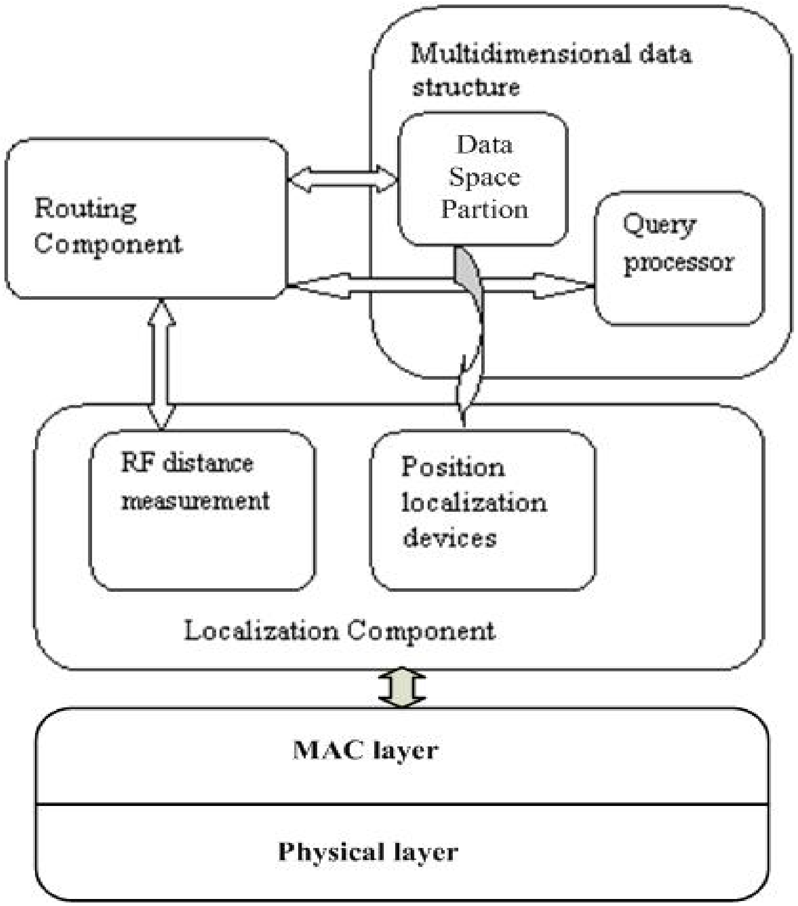
Directed Diffusion

- Data-centric and location-centric addressing
- Sinks express interest in some data attribute
- Interest is propagated (diffused) to desired location
- Data matching this attribute is reverse-propagated to sinks
- Passage of data refreshes gradients

Cross-Layer Integration and Optimization:

Localization, Data Storage, and Query Processing

Our integrated scheme - Architecture



Localization component finds the position by local interaction

Data space Partition uses online localization information to dynamically assign the corresponding data zone to the node

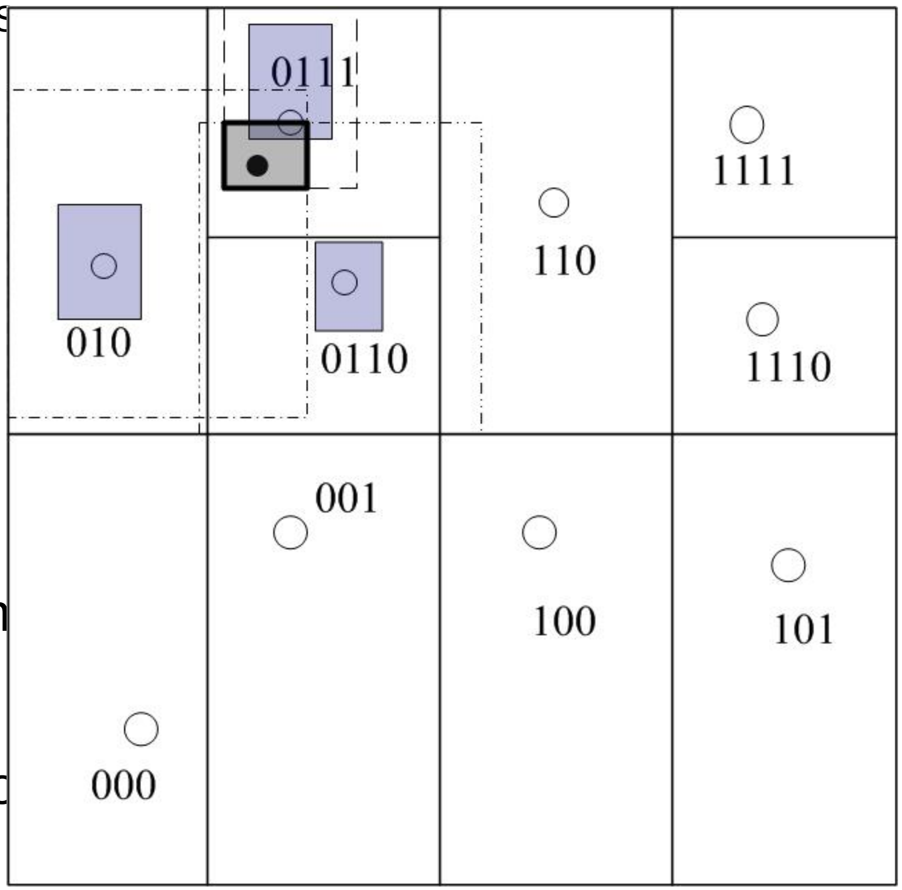
(The two are integrated in PRDS algorithm)

Routing component efficiently routs with high tolerance for errors.

Query processor handles query propagation, result collection, and performing various types of aggregations.

Localization-integrated indexing – PRDS algorithm

- Assuming a portion of nodes may have GPS or other geographical capabilities.
- Position Region (PR) -- PR(n) is calculated by the intersection of rectangles obtained from neighbor's ranging distances and their own location information. It may take several rounds to stabilize.
- Select the splitting partner with the largest position region coverage overlapping PR(n) and split that node's space into two with the joining node taking over one subspace.



Advantages

- Integration results in synergistic gain of efficiency
- PRDS requires O(deg) messages for each joining node. (deg being the node's out degree)
- Dynamic measurement of accumulated errors in the network
- Adaptation to changing environment.

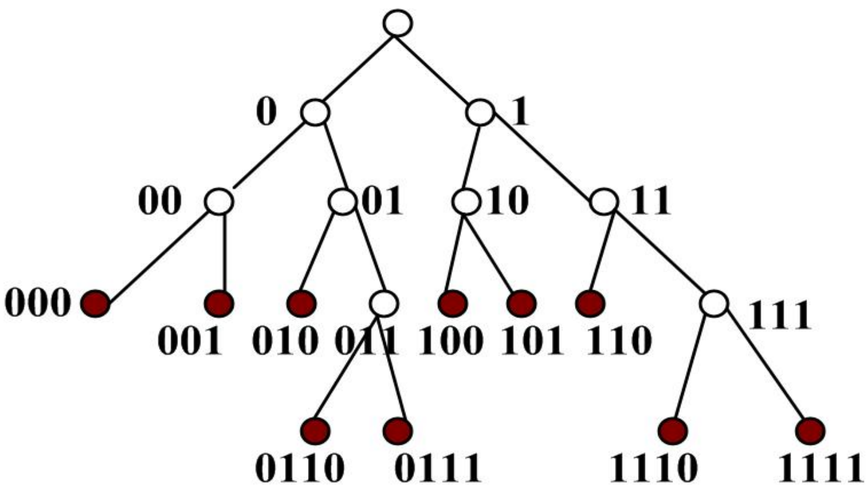
Query processing

- Single query
  - Calculate the identifier for the data to be retrieved
- Range query
  - Calculate the largest common prefix of the identifiers in the requested range.

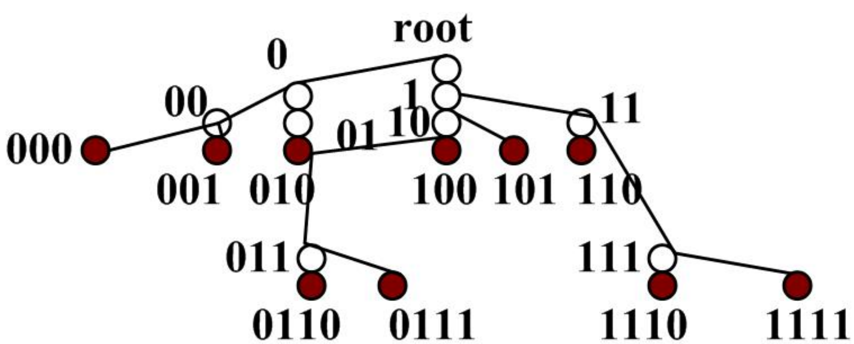
Aggregation

- In-network processing requires efficient structure
- Traditional approaches (TAG, Cougar etc)
  - Query flooding
  - Spanning tree generation for each query
- Our index tree approach
  - Avoids query flooding
  - Index tree is a by-product of underlying index structure, does not incur additional cost after setting up
  - Efficient performs aggregation throughout the network for arbitrary query issuer
  - Provide an easy platform for approximate queries

Aggregation (continued)



Identifier tree



Index tree

The space partition can be represented. by ***identifier tree***. The current zones are represented by the leaf nodes (colored black) in the identifier tree, derived from splitting the intermediate nodes.

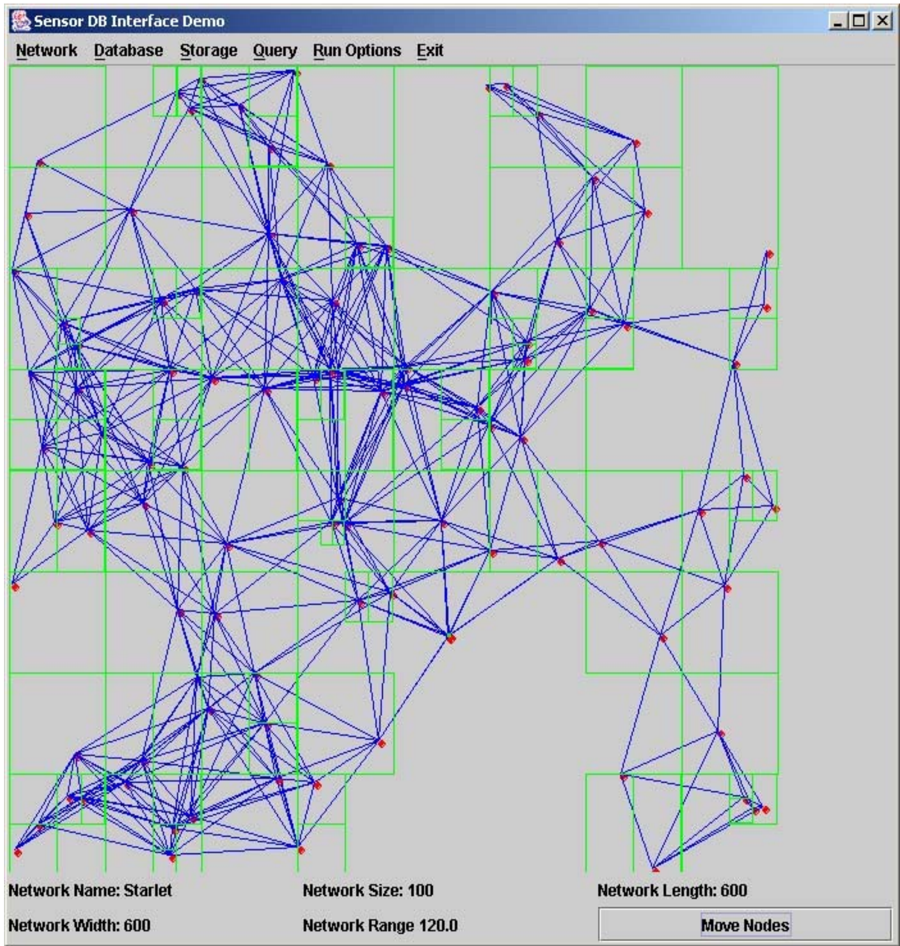
Index tree is constructed from identifier tree by assigning multiple responsibilities to some nodes that are going to serve as both intermediate node and leaf node in the aggregation process.

Aggregation root is discovered by calculating the root of the sub-tree it needs to explore from the scope of the aggregation.

Cost of index-tree based aggregations

	Min/Max	Count	Average	Sum
Routing cost	O(logN)	O(logN)	O(logN)	O(logN)
Sensor access	1	1	1	1
Summary update	No summary needed.	Yes.	No. computed by Sum/Count	Yes.
Update cost	N/A	O(logN) per item	N/A	O(logN) per item

Simulation platforms



TakeAways

- Sensor networks have the potential of assisting in many aspects of our life.
- Deploying and operating a large sensor network for long periods of time is not trivial.
- Energy-efficiency, fault-tolerance, security and privacy are important requirements for most sensor applications.
- These requirements should be taken into consideration in designing self-configurable sensor networks with query processing and storage capabilities.

Future Programming Paradigms

Distributed embedded computing

Software Challenges

- Support for large numbers of unattended device
  - Sensor nets vs. Internet?
- Adaptive behavior in an unpredictable environment
  - Sensor nets vs. automated manufacturing?
- Data centric communication

Programming Massively Distributed Systems

- Individual devices are not important
- Program must tolerate device failures and irregularity
- Program does not know exact device locations
- Program must provide the desired overall behavior

Growing Point Language

- Language abstractions:
  - Growing points
  - Pheromones

    ....

- Sequential conceptualization of a parallel computation

- Any planar design can be compiled into a GPL program

Programming in a Physical Environment

- In a sensor-rich environment, the human and machine can share the same physical model of the world
  - Physical objects can have a software representation
  - Physical object location can be part of software state
- New applications? New interfaces?

Active Bat System

- A smart-office system implemented by AT@T
- “Bats” are attached to tracked objects and people
- Bats emit ultrasonic signals
- An array of sensors in the building locates the bats by ranging
- Physical entities in the environment are represented as software CORBA objects

Conclusion

"*Knowledge of what is possible is the beginning of happiness*"

George Santayana (1863 - 1952)
US (Spanish-born) philosopher

Q&A

QUESTIONS

ANSWERS