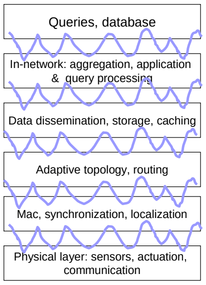
Layered Architecture

- Resource constraints dictate tightly coupled layers
- Can we define such a layered system for application-specific systems?



Queries, database

In-network: aggregation, application & query processing

Data dissemination, storage, caching

Adaptive topology, routing

Mac, synchronization, localization

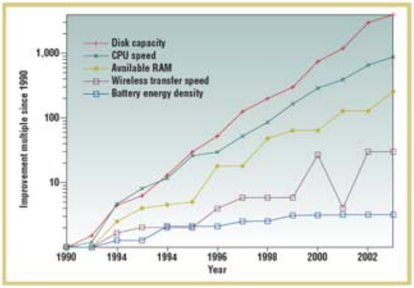
Physical layer: sensors, actuation, communication

Data-centric & In-network Processing

- Networks are/should be organized around the data/sensing phenomena
    - Node ID irrelevant
    - SN as a active database? Attribute-based addressing/routing
- No indirection in communication
    - Overhead is too high (No search engine, DNS, etc)
- Data correlation must be exploited in local neighborhoods
    - Can't scale
    - In-network processing is important (but not more important than making correct observations)

Challenge: Resource & Energy Constraints

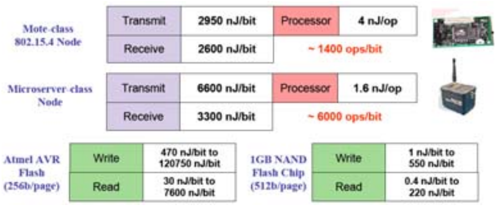
- Limited battery energy



Challenge: Resource & Energy Constraints

- Limited battery energy
    - Low-power platforms, energy harvesting
- Limited computing power, bandwidth, storage
    - Light-weight S/W, data centric protocols (diffusion)
- Capacity scaling
    - Exploit correlations in local neighborhoods
- Energy/bit communicated >> Energy/bit stored > Energy/bit processed
    - In-network processing to reduce # bits communicated
- Dominance of Receiving over Transmitting
    - Receiver (sleep more), low power listening
- Self-configuration
    - Time sync and node localization

Communication vs. Storage vs. Processing

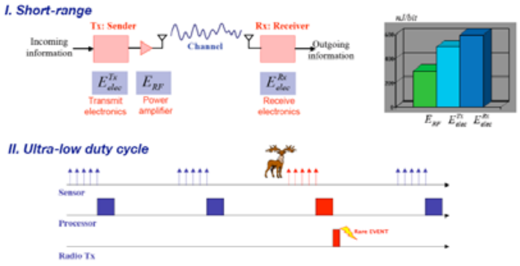


Mote-class 802.15.4 Node	Transmit	2950 nJ/bit	Processor	4 nJ/op
	Receive	2600 nJ/bit	~ 1400 ops/bit	
Microserver-class Node	Transmit	6600 nJ/bit	Processor	1.6 nJ/op
	Receive	3300 nJ/bit	~ 6000 ops/bit	

Atmel AVR Flash (256b/page)	Write	470 nJ/bit to 120750 nJ/bit	1GB NAND Flash Chip (512b/page)	Write	1 nJ/bit to 550 nJ/bit
	Read	30 nJ/bit to 7600 nJ/bit		Read	0.4 nJ/bit to 220 nJ/bit

Energy/bit sent >> Energy/bit stored > Energy/op

Radio: Receive/Idle over Transmit



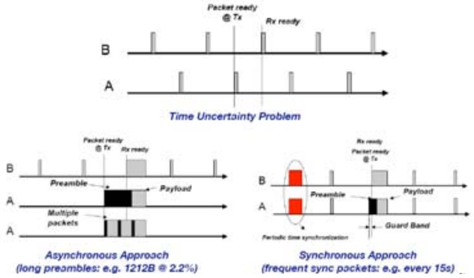
Optimize for Receive Energy

•S-MAC, STEM, T-MAC, B-MAC, RATS/U-BMAC, etc

Difference from traditional distributed systems

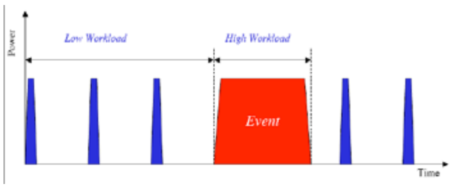
- It's all about ENERGY
  - ☐ Listening to the network is no longer free
- Distributed systems never had to deal with correlated data
  - ☐ Routing structure
  - ☐ Data correlation
  - ☐ In-network data processing
- Much more severe constraints
  - ☐ A typical internet application is conceived with unlimited power, 10s to 100s of Mbps bandwidth, TeraBytes of storage…
- Self-organized, intelligent, autonomous systems
  - ☐ No IP addresses, DNS etc

Time = Energy

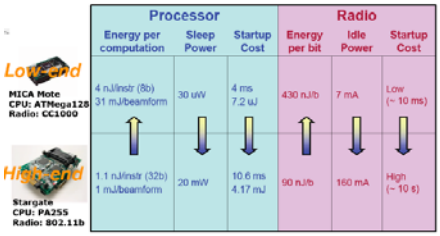


Srivastava '08

Low Power vs. Performance & Energy



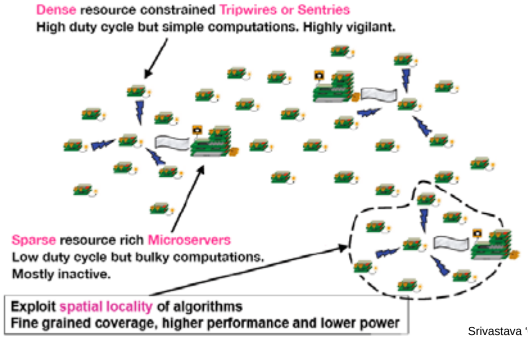
Low Power vs. Performance & Energy



	Processor			Radio		
	Energy per computation	Sleep Power	Startup Cost	Energy per bit	Idle Power	Startup Cost
Low-end MICA Mote CPU: ATMega128 Radio: CC1000	4 nJ/instr (8b) 31 mJ/beamform	30 uW	4 ms 7.2 uJ	430 nJ/b	7 mA	Low (~ 10 ms)
High-end Stargate CPU: PA255 Radio: 802.11b	1.1 nJ/instr (32b) 1 mJ/beamform	20 mW	10.6 ms 4.17 mJ	90 nJ/b	160 mA	High (~ 10 s)

- Adaptive Architectures
  - ☐ Vary H/W resources, eg. PASTA, LEAP

Tiered Architectures



Dense resource constrained Tripwires or Sentries
High duty cycle but simple computations. Highly vigilant.

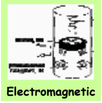
Sparse resource rich Microservers
Low duty cycle but bulky computations.
Mostly inactive.

Exploit spatial locality of algorithms
Fine grained coverage, higher performance and lower power

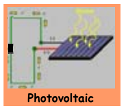
Srivastava '08

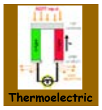
Energy Harvesting


Piezoelectric


Electromagnetic


Underwater piezo-eel

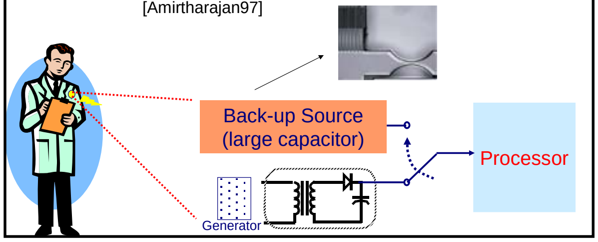

Photovoltaic


Thermoelectric

Harvesting technology	Power density
Solar cells (outdoors at noon)	$15mW/cm^2$
Piezoelectric (shoe inserts)	$330\mu W/cm^3$
Vibration (small microwave oven)	$116\mu W/cm^3$
Thermoelectric (10^oC gradient)	$40\mu W/cm^3$
Acoustic noise (100dB)	$960nW/cm^3$

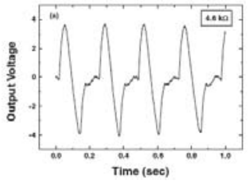
Example: Self-powered Chip

- Power generated using motion or solar cells, and stored in a backup source (e.g. large capacitor)
  - no batteries needed
  - applicable to sensors on vehicles, body etc.
  - e.g. *Embedded power supply processor* from MIT [Amirtharajan97]

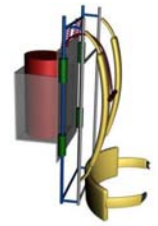


Example: Sensor Nodes with Windmills

- Sensor nodes with windmills of diameter 10 cm attached to a rotating cam that flexes a series of piezoelectric crystals as it rotates, causing them to generate a current
  - A 'gentle breeze' of 16km/h is enough to generate the 7.5mW
- Ref: Piezoelectric Windmill: A Novel Solution for Remote Sensing, Priya et. al., Japanese Journal of Applied Physics, 2005.
  - http://mse.uta.edu/Priya/Piezoelectric%20Windmill.pdf



Vibration Energy Harvesting



Experimental device for investigating displacements and forces between backpack and individual

Objective
To optimization of electostrictive materials for the generation of electrical energy from human motion

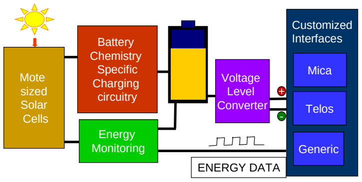
Approach
Electrostrictive polymers/composites will be custom designed to provide optimal matching with both the forces and displacements associated with various types of natural human motion

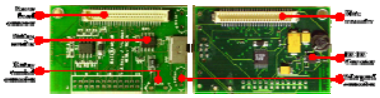
Payoff
Supplementary and/or emergency electrical energy for individual Marines.

Principal Investigators
Penn State
University of Pennsylvania

Courtesy: US Dept of the Navy, Office of Naval Research, Project #353 expeditionary, 2005

Example Energy Harvesting Node: *HelioMote*



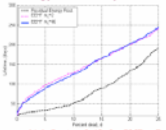


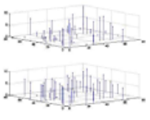
Energy Harvesting

- Current State
  - Mote with simple sensors, periodic sampling @ low rate, low duty cycle (< 1%) : 1 year
  - Microserver with high rate sensors: 1 week
- Harvesting-aware nodes
  - 20+ years @ 20-60% duty cycle
- **Energy neutral operation**

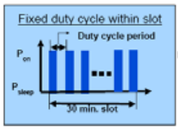
Energy Harvesting Aware Management

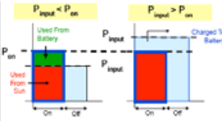
A. Routing: *link cost function of energy-availability*

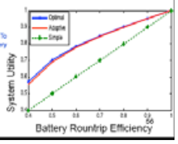




B. Duty cycling: *predict energy-availability slot-by-slot and adapt*







Sensor are more Demanding

- Focus on communication
  - ☐ Simple applications, low-rate, low complexity sensors (light, temp, etc..)
- Emerging applications
  - ☐ Energy hungry sensing (video, cameras, acoustic arrays, precision sensors, etc)
  - ☐ Actuators
    - Robotic nodes, Directional antennas/cameras
- Need to optimize for sensor sampling

Self-Configuration Crucial

Ad Hoc Deployment

Rapid Infrastructure Setup

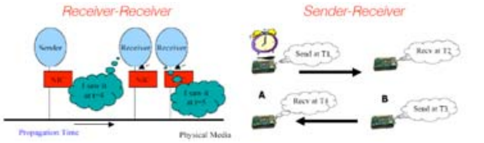
Unknown Obstacles in Environment

- Time Synchronization
- Node localization
- Calibration

Self-Configuration crucial to relate data to real-world

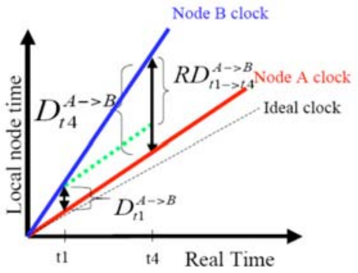
Time Synchronization

- Basic Approach
  - ☐ Measure clock offsets between neighbor nodes
  - ☐ Use offset to measure drift
  - ☐ Use drift and latest measured offset to estimate event occurrence



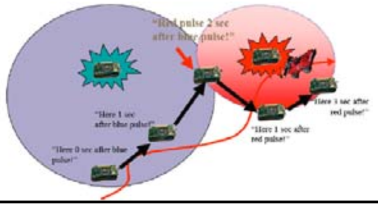
Instantaneous offset estimation of 5-10µs (Mica2)

Clock Drift Estimation

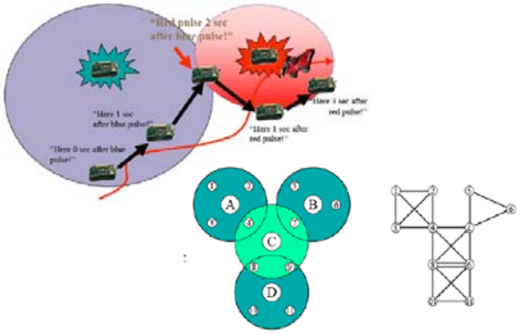


Multi-hop Time Synchronization

- Variant
  - ☐ Global sync not local
  - ☐ Absolute time vs relative
  - ☐ Proactive vs post event
- 2-steps
  - ☐ Establish pairs with offsets
  - ☐ Accumulate offsets along forwarding path

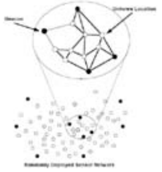


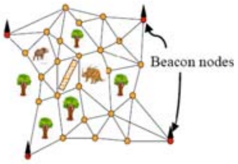
Multi-hop Time Synchronization



Localization

- Absolute: use beacons to give a proximity frame-work
- Relative: localize relative to each other in a locally defined frame





Localization

- Establish a geometric relationship
  - Distance
    - Radio link quality/signal strength
    - Acoustic/Radio time of flight with atomic clocks
    - Radio RTT
    - Vision based
  - Direction
    - RF, IR, acoustic directional signal
    - Vision based
- Algorithms fro estimating location
  - Triangulation
  - Multilateration
  - Track over time
- Examples: Cricket, AhLoS, RIP

Current Transition: Now

- “Smart Dust” (1000+ devices)
  - Minimize each node resources
  - Exploit numbers
- Fully autonomous systems
- In-network & collaborative processing
- Optimize communication (energy)

Current Transition: Future

- Heterogeneous systems
  - Tiered
  - Under-sampling (time/space)
  - Multiple modalities, scales, mobility
- Interactive systems
  - Online interaction, tasking
- In-network & collaborative processing
  - Data quality, data control, responsiveness
- Monitor the monitors: calibration, self-test, validation, security