Introduction to Sensor Networks

Maria Potop-Butucaru, Franck Petit, Sébastien Tixeuil
Sorbonne Universite, France
Firstname.Lastname@lip6.fr

Outline

- Motivation
- Architecture
- Overview

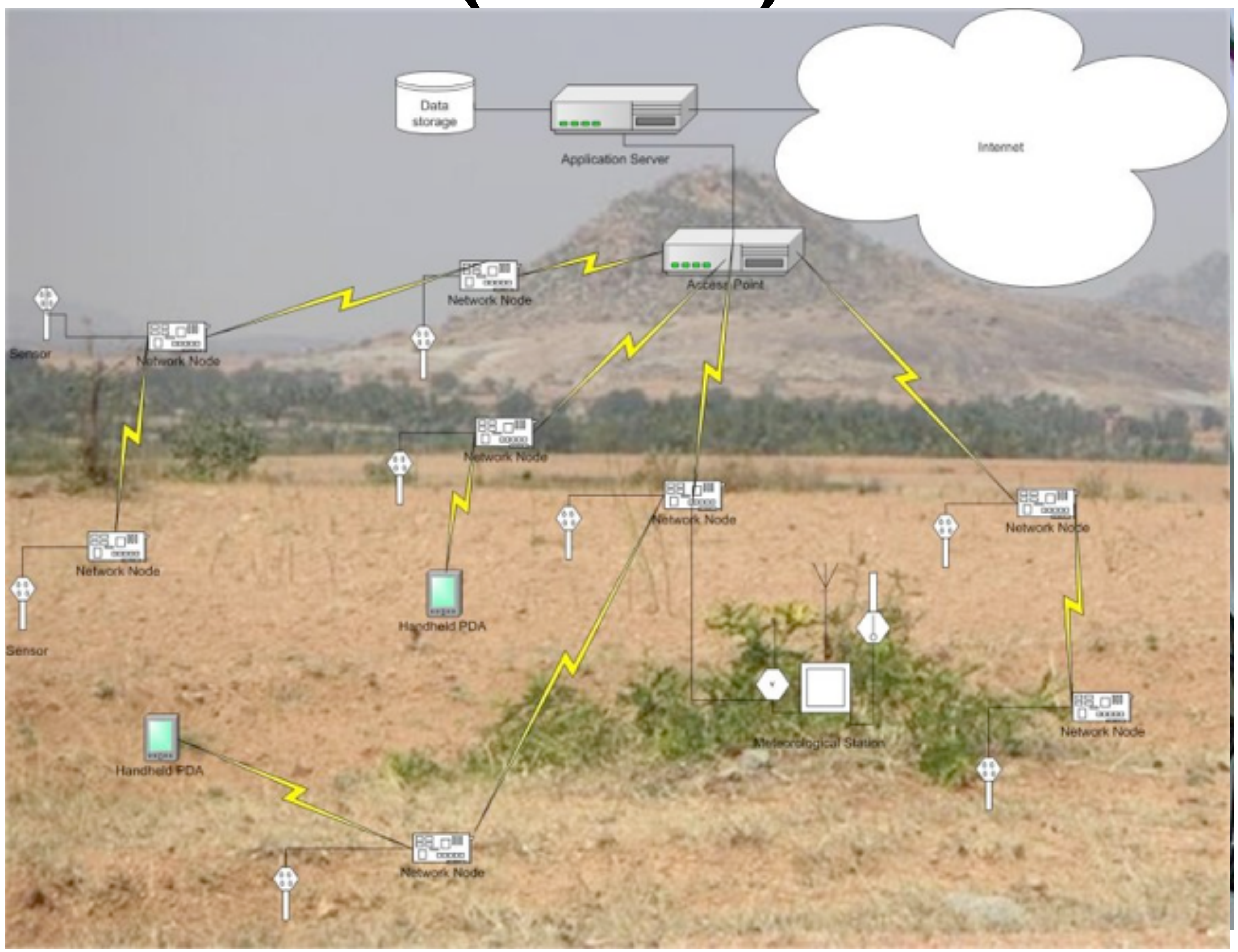
Motivation

Sensor Networks

- **Definition**: Network of wireless nodes dedicated to a particular application
- **Purpose**: Acquire sensed data and transmit to a processing station
- **Application domains**: Military, Civilian, Environment, Wildlife, etc.

Applications

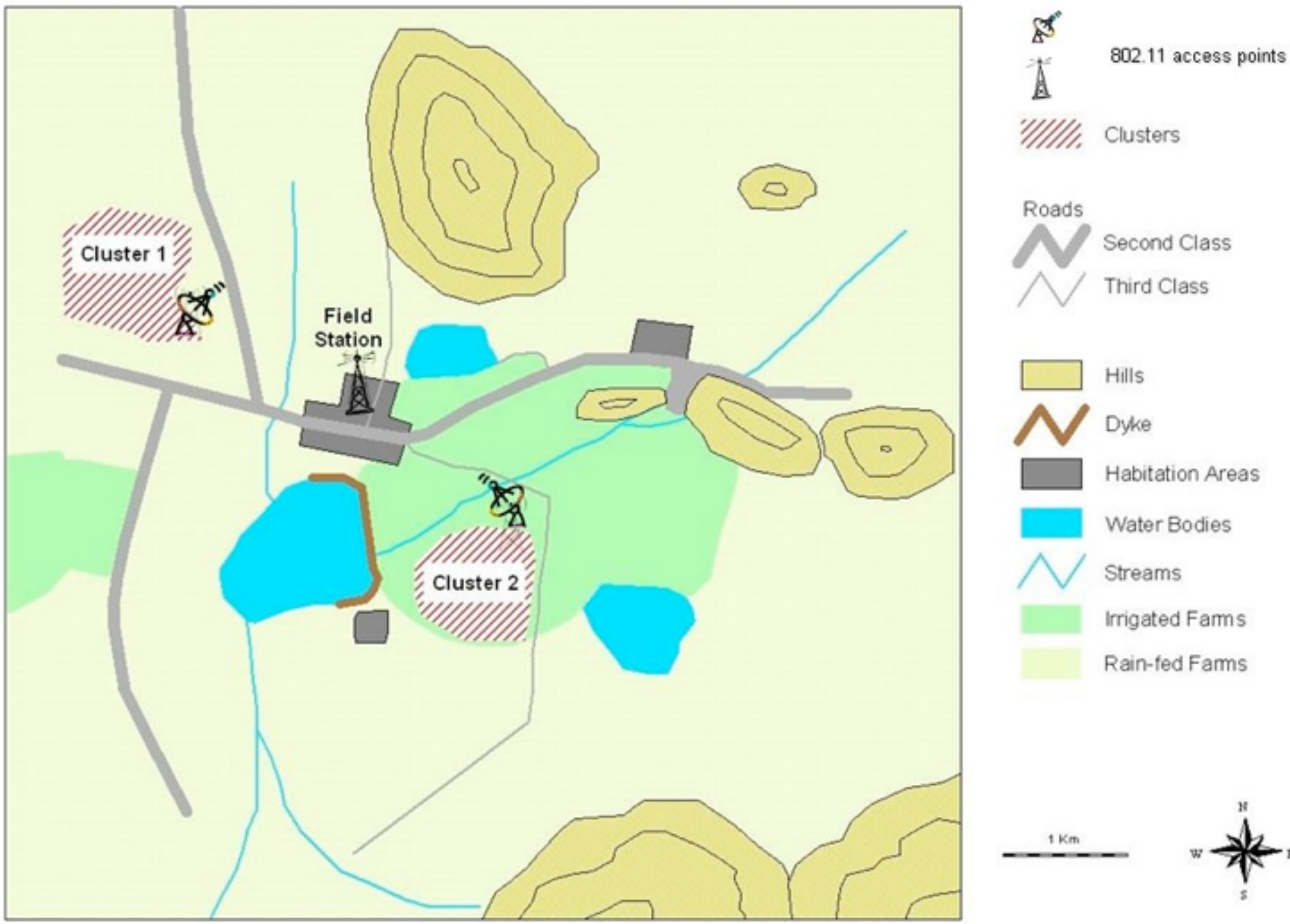
COMMON-Sense Net (CSN)



Tumkur, Karnataka

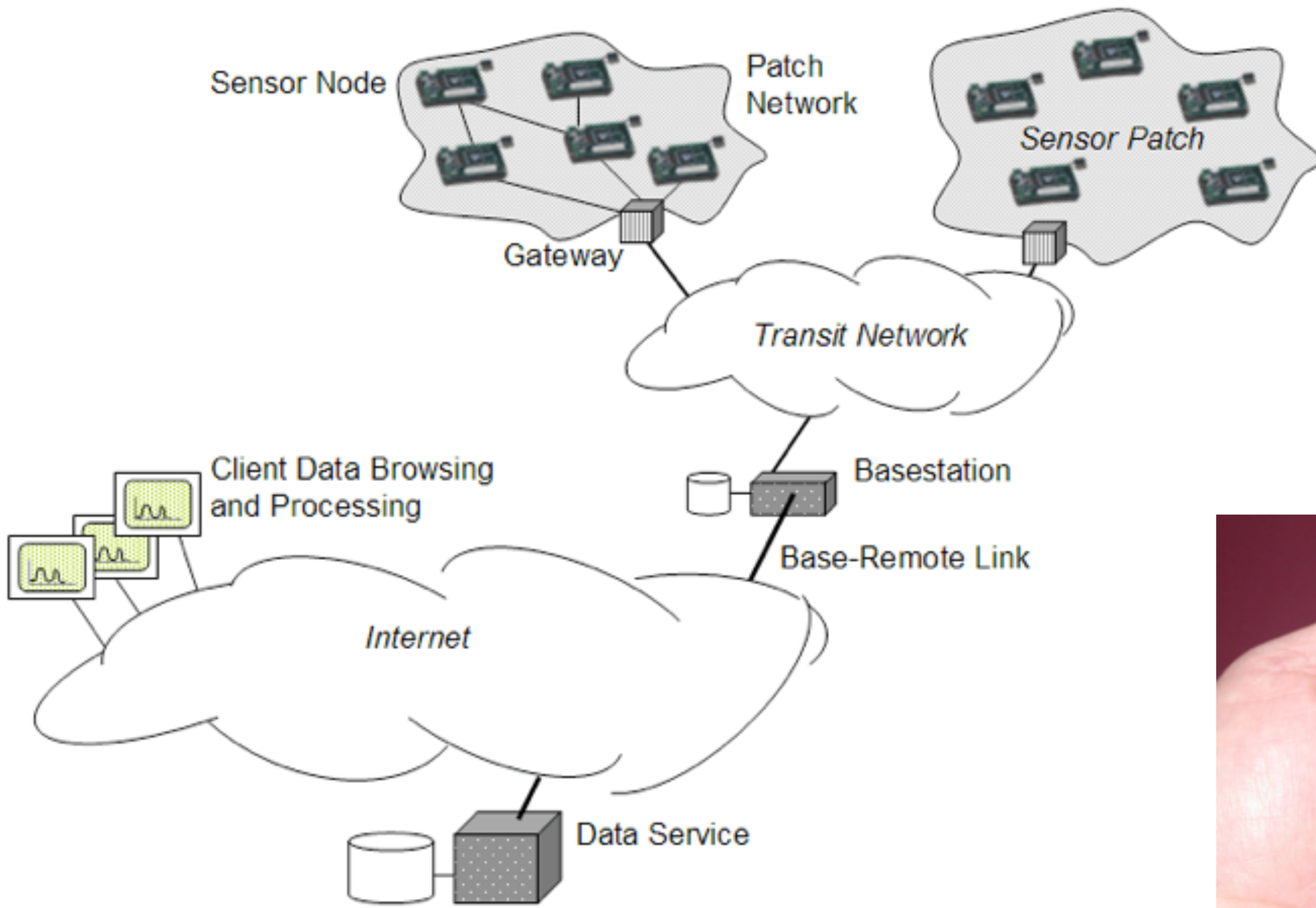
http://commonsense.epfl.ch/

COMMON-Sense Net (CSN)

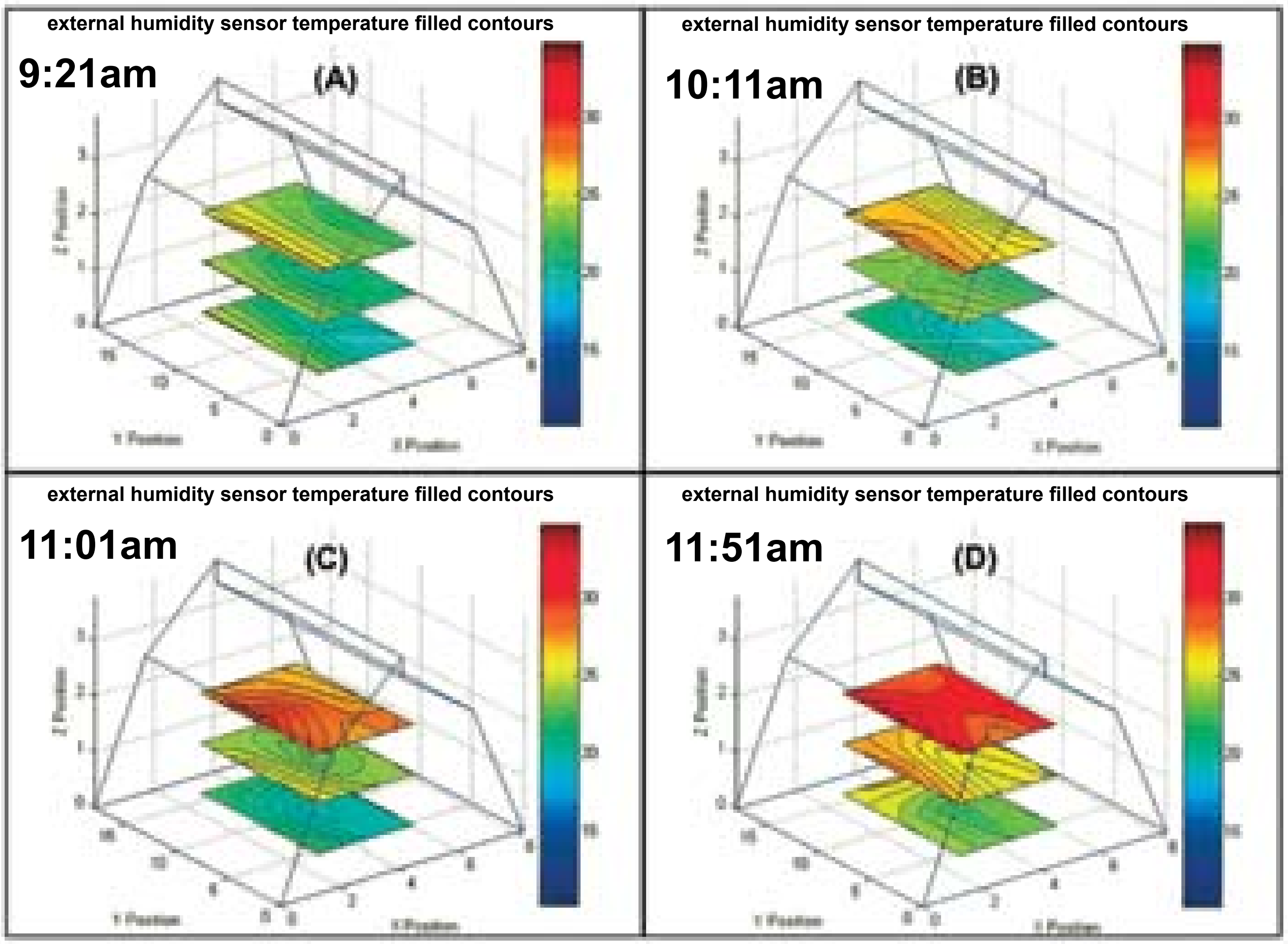


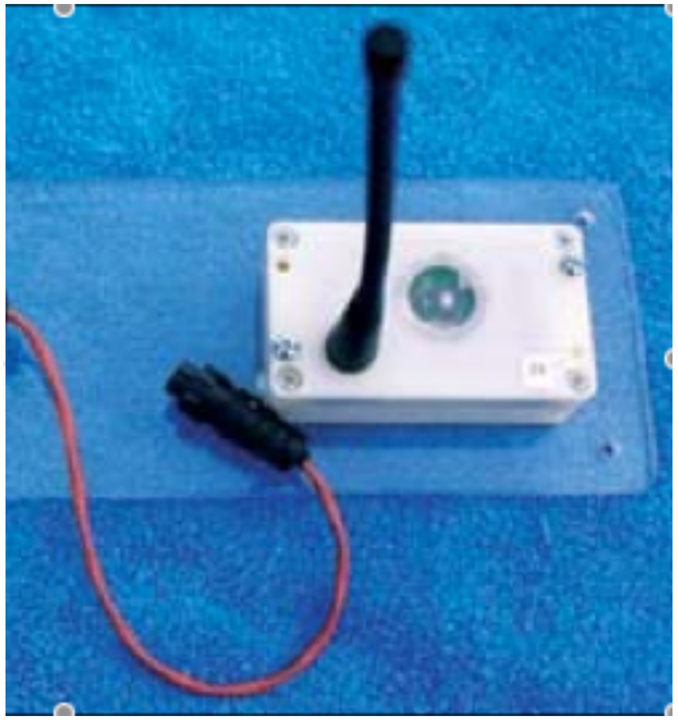
http://commonsense.epfl.ch/

Habitat Monitoring (Berkeley)

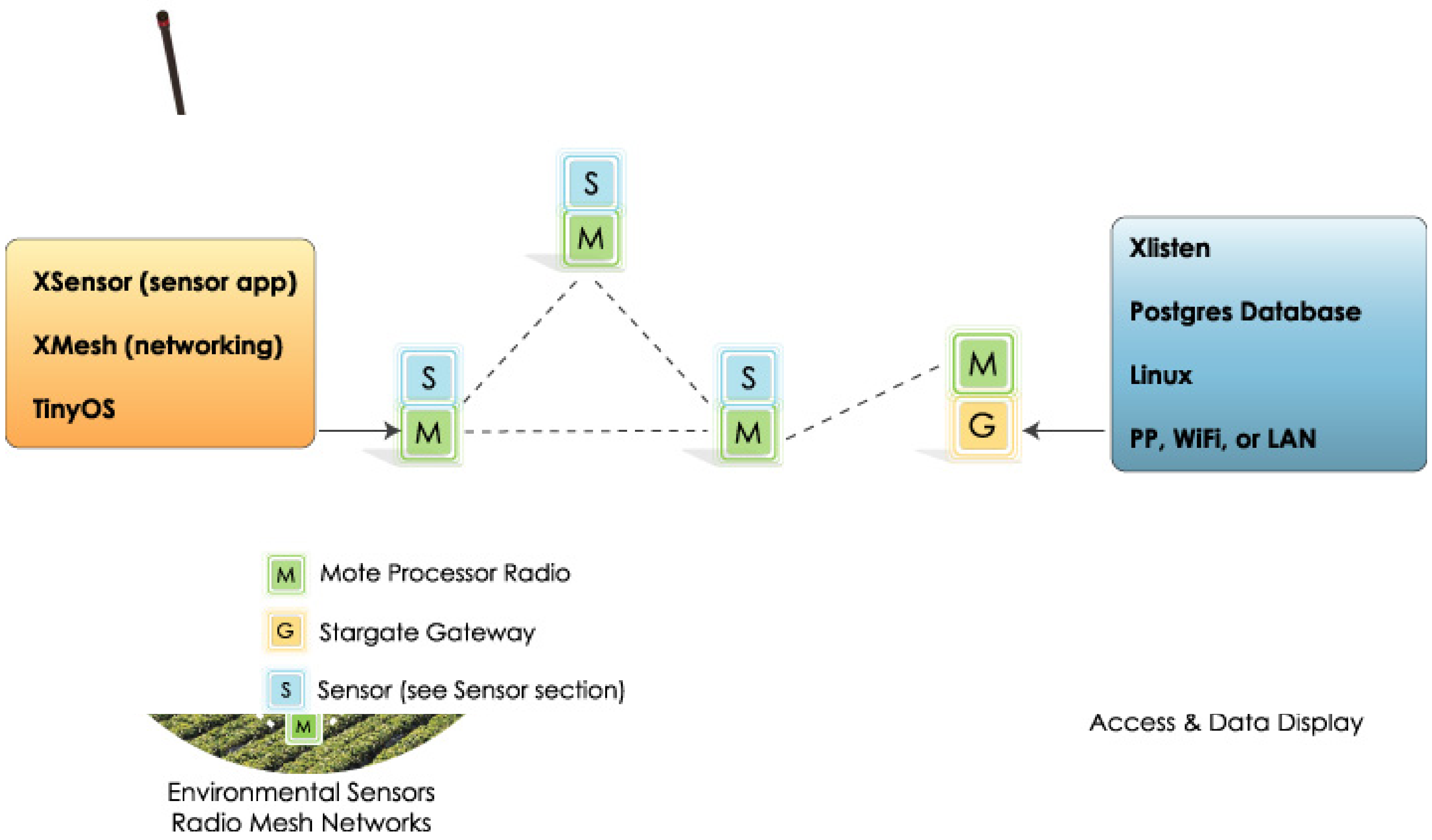


Habitat Monitoring

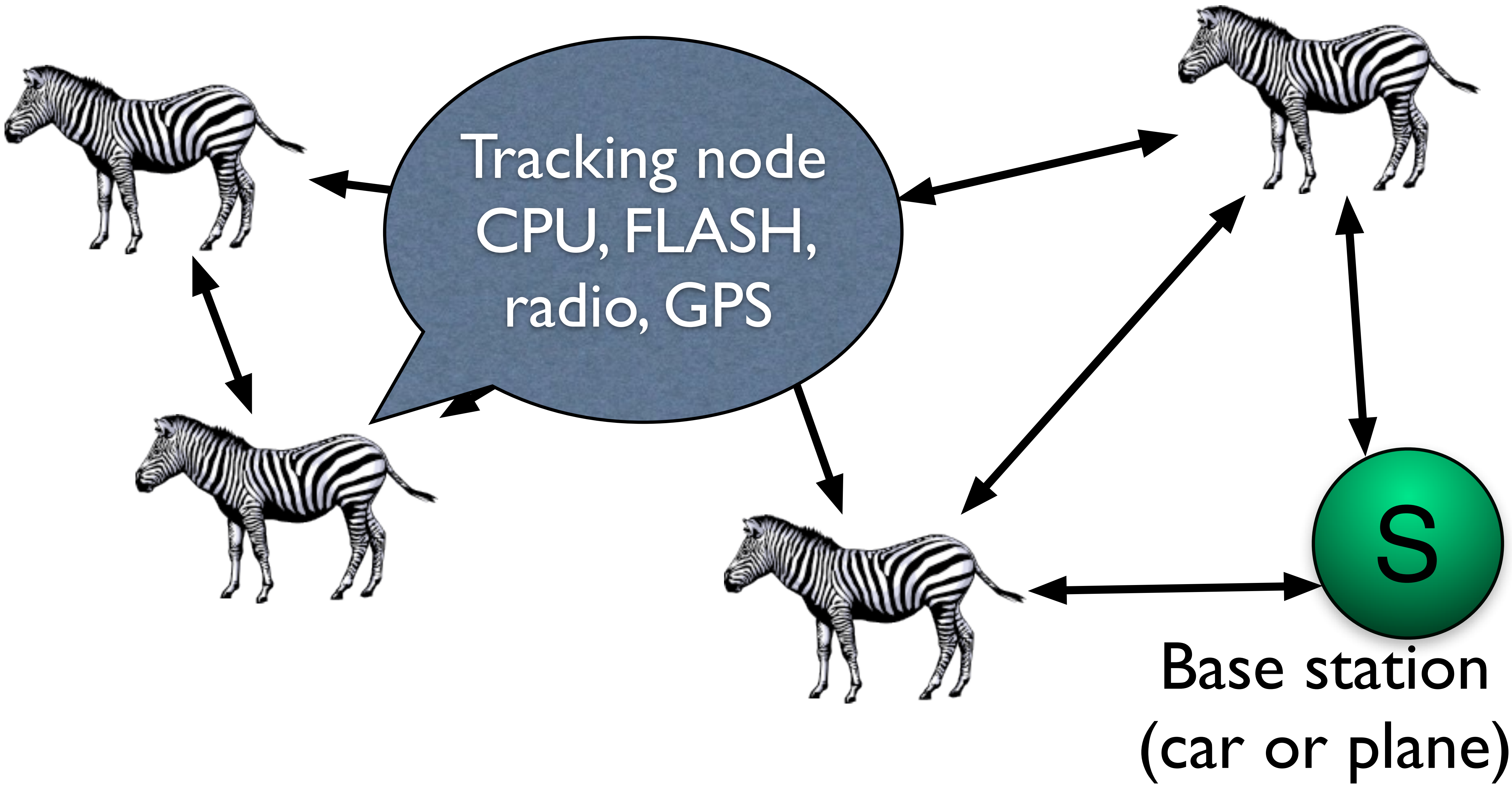




Environment Monitoring



Zebranet (Princeton)



Sweetwaters Reserve, Kenya

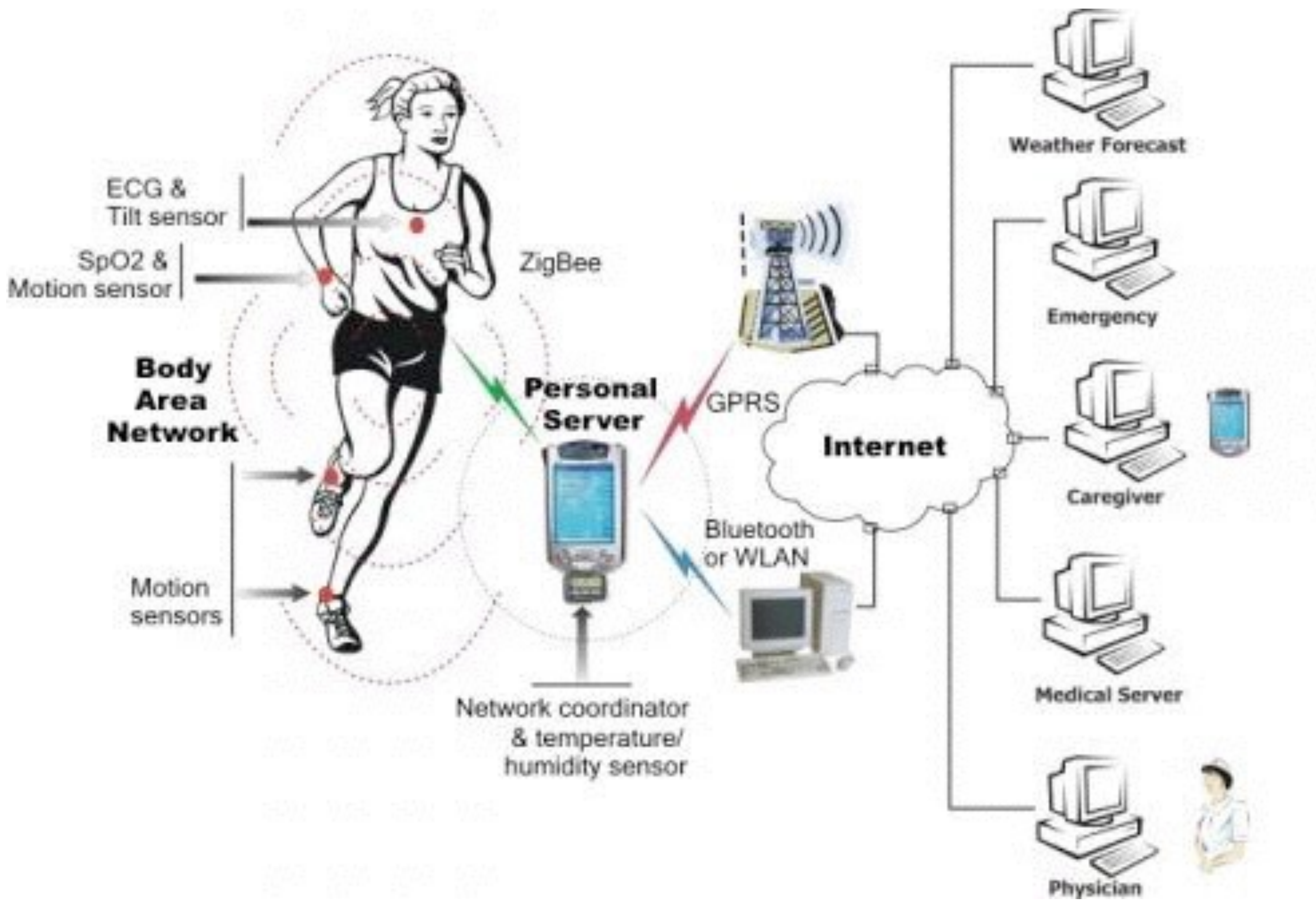
Zebranet

Attribute	Zebranet	Sensors
Mobility	High	Low/static
Range	Miles	Meters
Frequency	Constant	Sporadic
Power	Hundreds of mW	Tens of mW

Zebranet

Sweetwaters Reserve, Kenya

WBAN



ECG &
Tilt sensor
SpO2 &
Motion sensor
ZigBee
Body
Area
Network
Motion
sensors
Personal
Server
Network coordinator
& temperature/
humidity sensor
GPRS
Bluetooth
or WLAN
Internet
Weather Forecast
Emergency
Caregiver
Medical Server
Physician

WSN *vs.* WBAN

	WSN	*WBAN*
Number of sensors	Important	Small (5-7)
Covering Area	Large (m-km)	Small (cm)
Standards	802.11	802.15.4 - .6
Network Topology	Not controlled	Controlled
Security	Medium	Important

Motivation

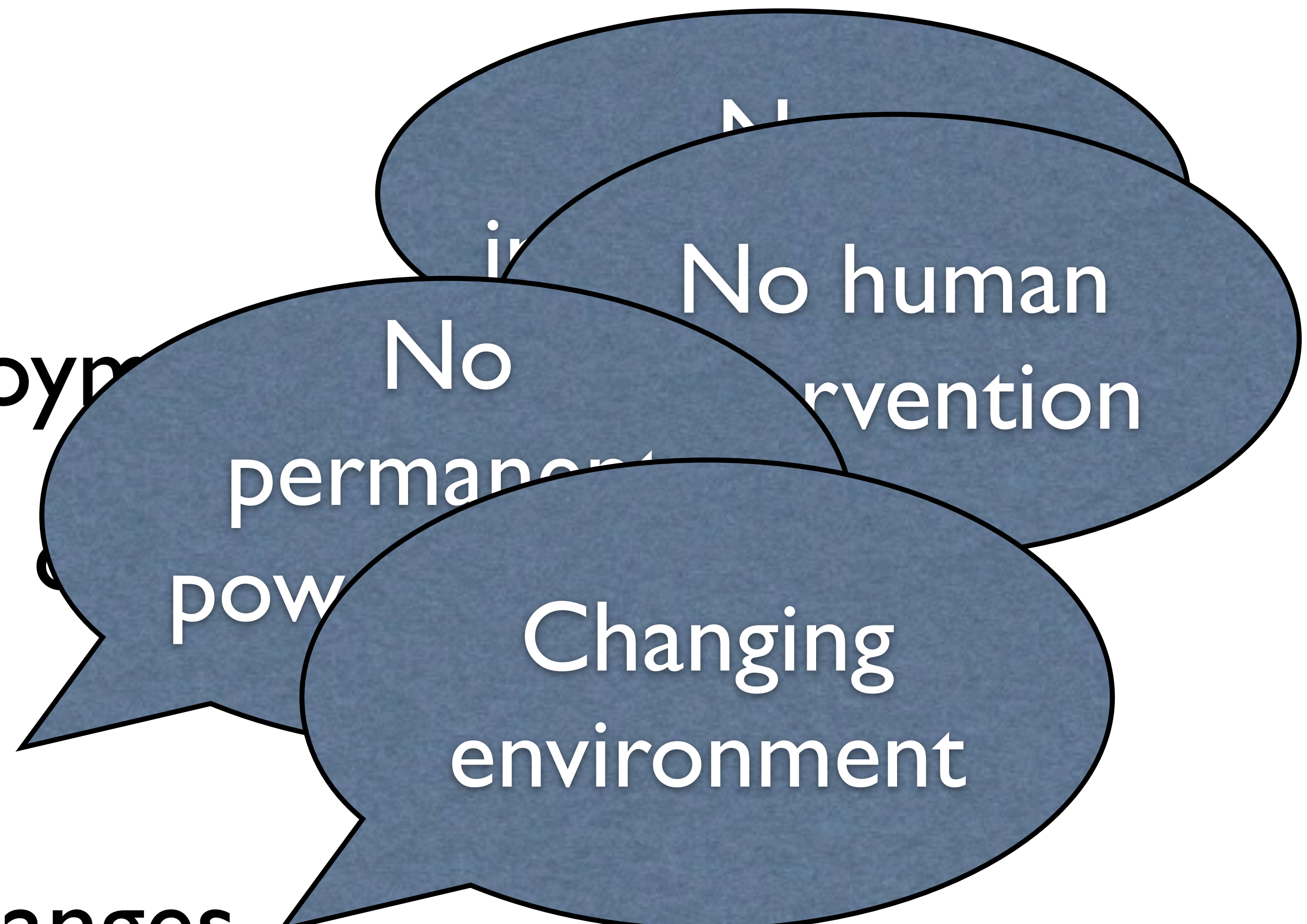
- **Acquire data and feed a processing station**
- **Application domains:**
  - *Military*: risky area monitoring, intrusion detection, etc.
  - *Civilian*: fire detection, chemical facilities monitoring, etc.

Sensor *vs. ad hoc*

Sensors	*ad hoc*
Specific	Generic
Collaboration	Selfishness
Many-to-one	Any-to-any
No ID	ID
Energy	Throughput

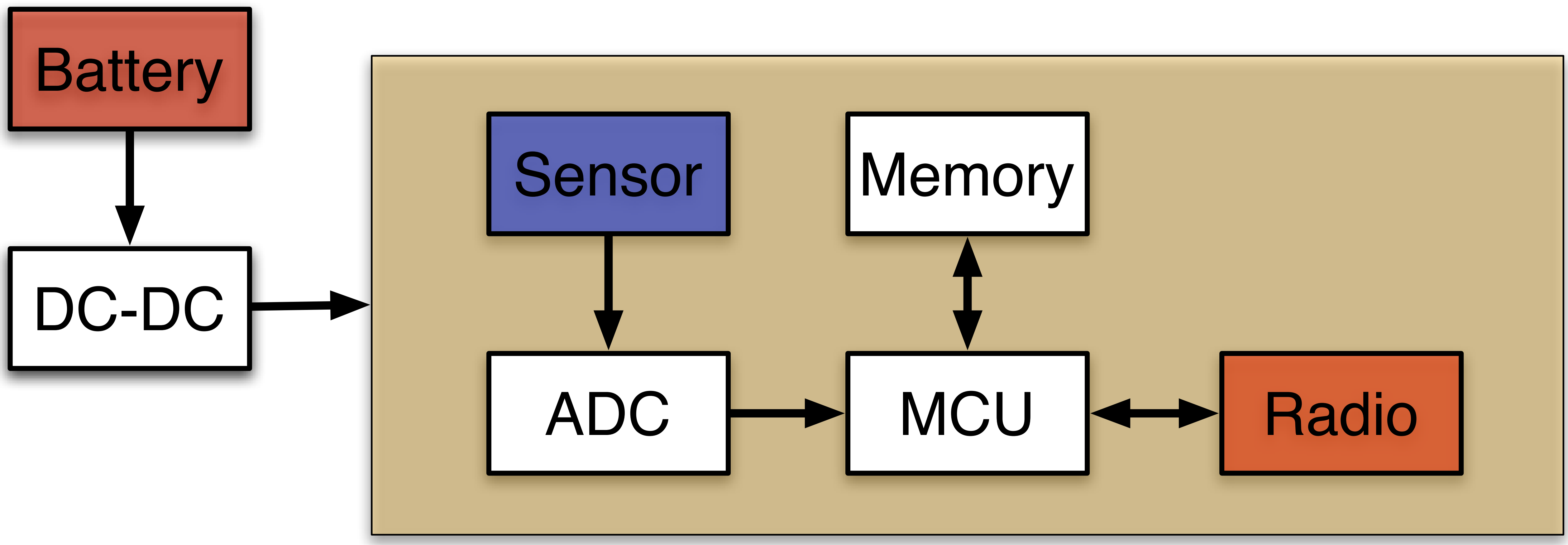
Issues

- *ad hoc* deploym
- Unattended c
- Untethered
- Dynamic changes



Architecture

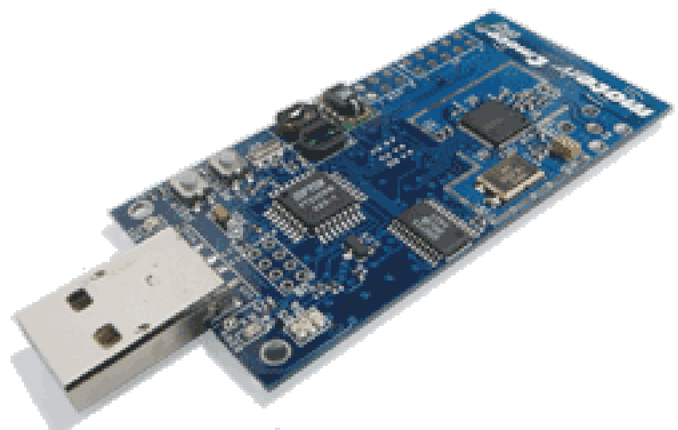
Sensor Node Architecture



Available Devices

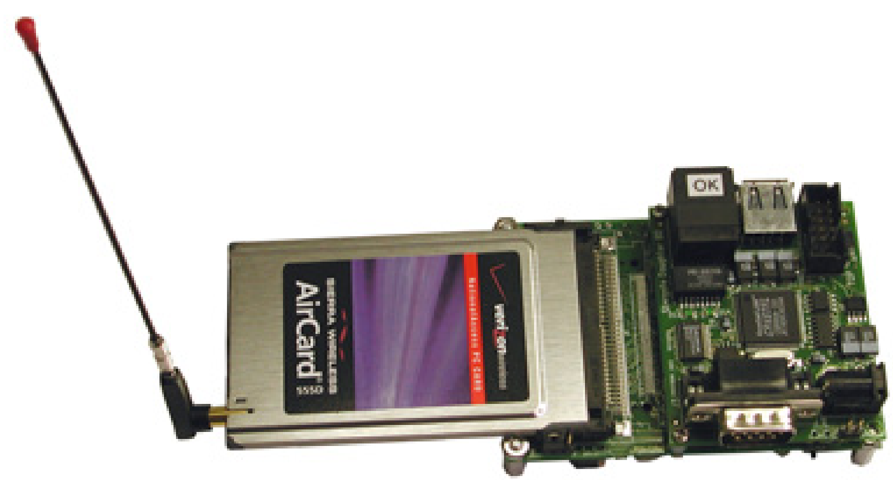
- MicaZ (Crossbow)
- 2.94 GHz IEEE 802.15.4 Zigbee radio
- 128 KB program memory
- 512 KB data memory
- 8 mA draw

Available Devices



- Tmote Sky/invent (Moteiv)
- 2.94 GHz IEEE 802.15.4 Zigbee radio
- 8MHz processor
- 10 KB RAM
- 48 KB Flash

Available Devices



- Stargate (Crossbow)
- Wired Ethernet
- Wifi/Cellular via PCMCIA
- INTEL PXA 255
- Linux Kernel

Available Devices

- GreenNet (ST Micro)
- 802.15.4 radio
- Solar energy harvesting

Sensors

- **Exterioceptors**: information about the surroundings
- **Proprioceptors**: information about the internal workings

Sensors

	Measurand	Transduction
Physical	Pressure	Piezoresistive, capacitive
	Temperature	Thermistor, thermomechanical, thermocouple
	Humidity	Resistive, capacitive
	Flow	Pressure change, thermistor

Sensors

	Measurand	Transduction
Motion	Position	E-mag, GPS, contact
	Velocity	Doppler, Hall effect, optoelectronic
	Angular velocity	Optical encoder
	Acceleration	Piezoresistive, piezoelectric, optical fiber

Sensors

	Measurand	Transduction
Contact	Strain	Piezoresistive
	Force	Piezoelectric, piezoresistive
	Torque	Piezoresistive, optoelectronic
	Vibration	Piezoresistive, piezoelectric, optical fiber, sound, ultrasound

Sensors

	Measurand	Transduction
Presence	Tactile	Contact switch, capacitive
	Proximity	Hall effect, capacitive, magnetic, seismic, acoustic, RF
	Distance	E-mag (sonar, radar, lidar), magnetic, tunelling
	Motion	E-mag, IR, acoustic, seismic

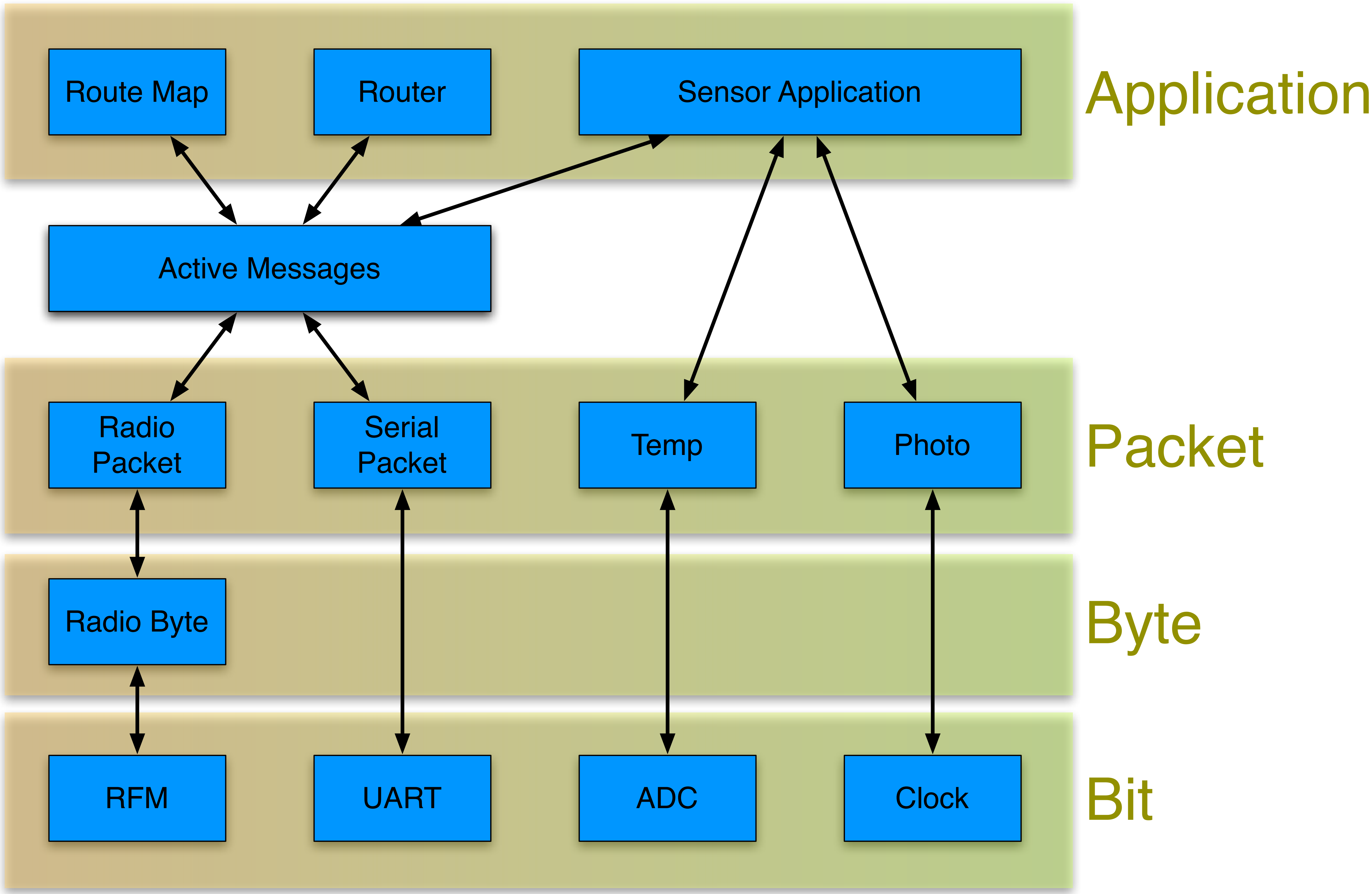
Sensors

	Measurand	Transduction
Biochemical	Agents	biochemical transduction
Identification	Personal features	Vision
	Personal ID	Fingerprints, retinal scan, voice, heat plume, vision, motion analysis

Sensor Node Operating System

- **TinyOS concepts**
  - Scheduler + Graph of components
  - Component
  - Constrained storage model
  - Very Lean multithreading
  - Efficient Layering

TinyOS Application



Programming TinyOS

- **TinyOS is written in NesC**
  - Applications are written as system components
- **Syntax for concurrency and storage model**
- **Compositional support**
  - Separation of definition and linkage

Simulating TinyOS

- **Target platform: TOSSIM**
  - Native instruction set
  - Event driven execution mapped to event drivent simulator
  - Storage model mapped to virtual nodes
  - Radio and environmental models

Other OSes for WSN

- **MagnetOS**
  - Virtual machines, byte code
- **Mantis**
  - Pure multithread
- **Contiki**
  - Dynamic linking of binaries
  - Event/Thread hybrid

Overview

Issues and Solutions

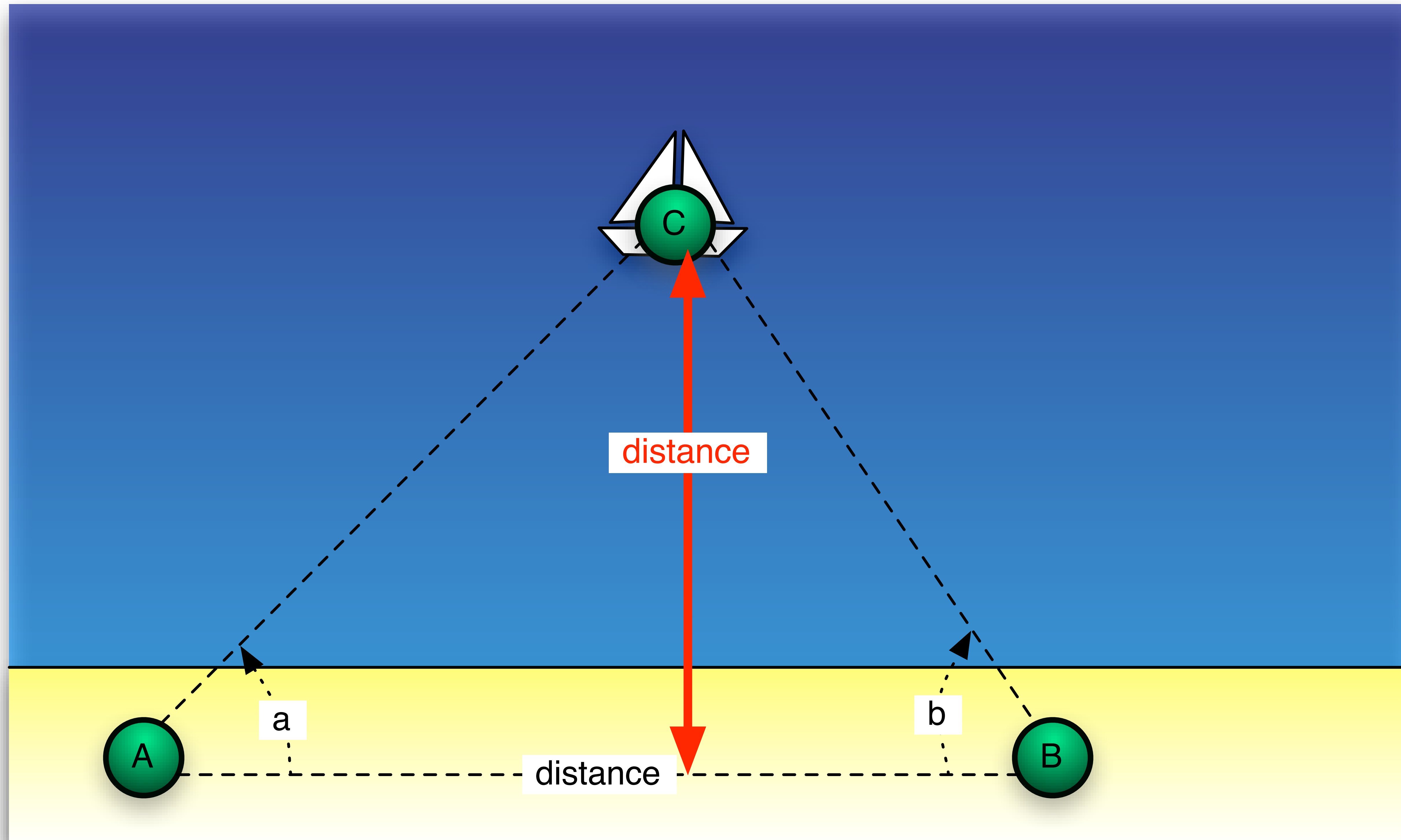
- Localization
- Routing
- Medium Access Control
- Applications

Localization

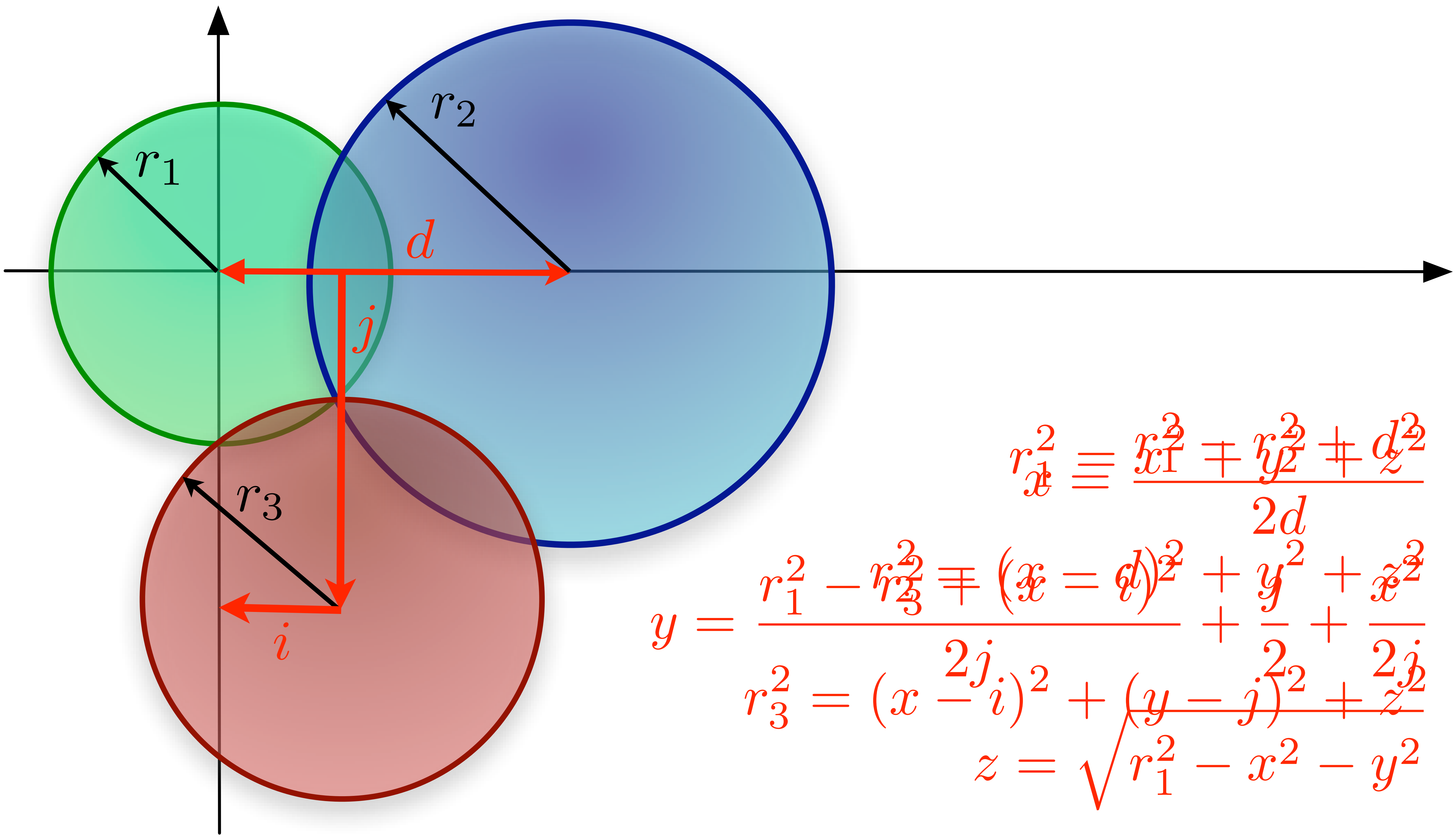
Localization

- **Fine-grained**
  - Timing
  - Signal strength
  - Signal pattern matching
  - Directionality
- **Coarse-grained**

Triangulation



Trilateration



$$r_1^2 = x^2 + y^2 + z^2$$

$$x = \frac{r_1^2 - r_2^2 + d^2}{2d}$$

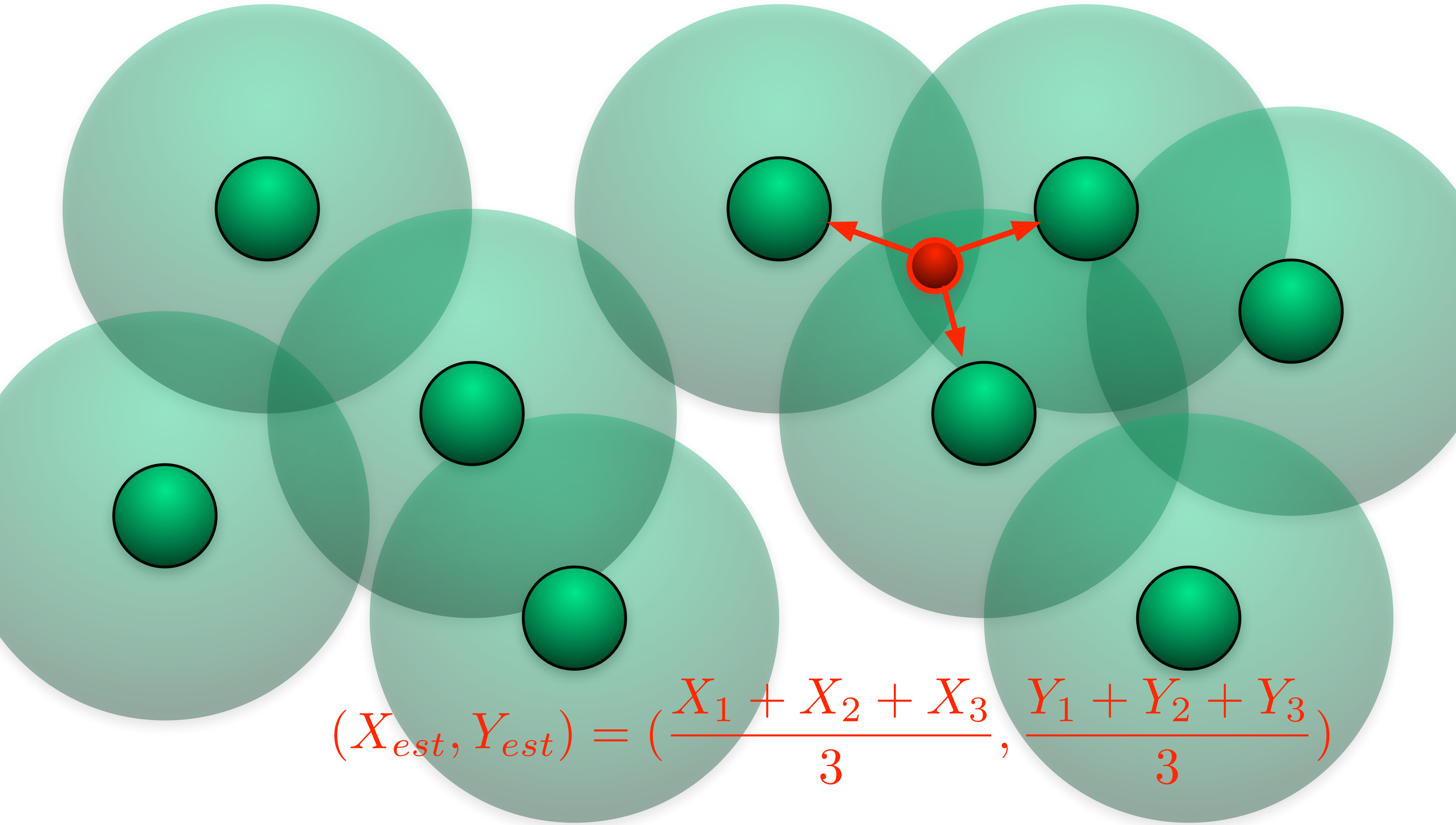
$$r_2^2 = (x-d)^2 + y^2 + z^2$$

$$y = \frac{r_1^2 - r_3^2 + (x-i)^2}{2j} + \frac{j}{2} - \frac{x^2}{2j}$$

$$r_3^2 = (x-i)^2 + (y-j)^2 + z^2$$

$$z = \sqrt{r_1^2 - x^2 - y^2}$$

Centroid



$$(X_{est}, Y_{est}) = (\frac{X_1 + X_2 + X_3}{3}, \frac{Y_1 + Y_2 + Y_3}{3})$$

Routing

Routing

- **Classical flooding**
  - *Implosion*
  - *Resource management*
- **Negociation based protocols**
  - *SPIN*
  - *Directed Diffusion*

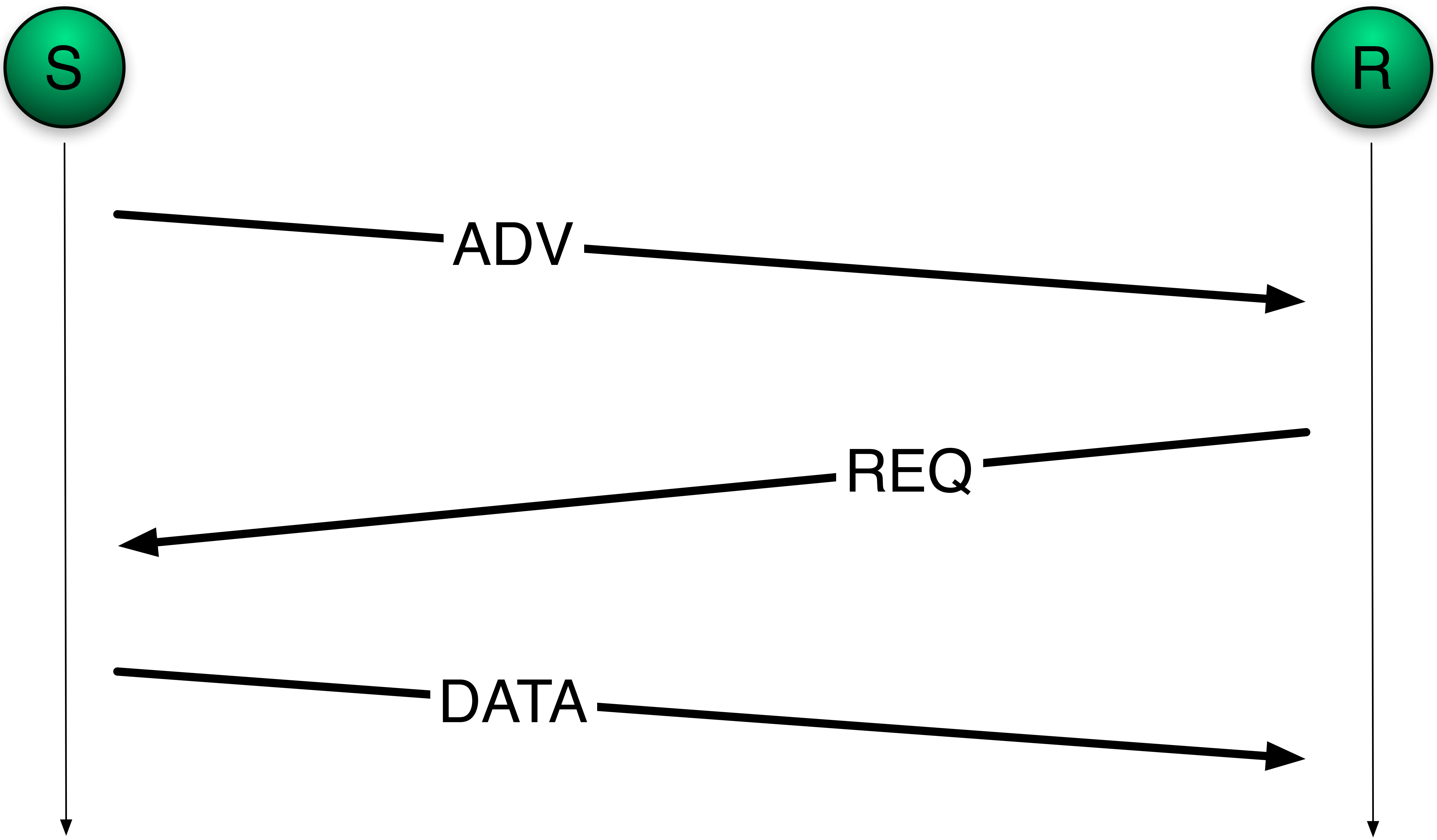
Negociation Based Protocols

- **SPIN**: Sensor Protocols for Information *via* Negociation
  - Information descriptors for negociation prior to data transmission
  - Negociation relates to available energy

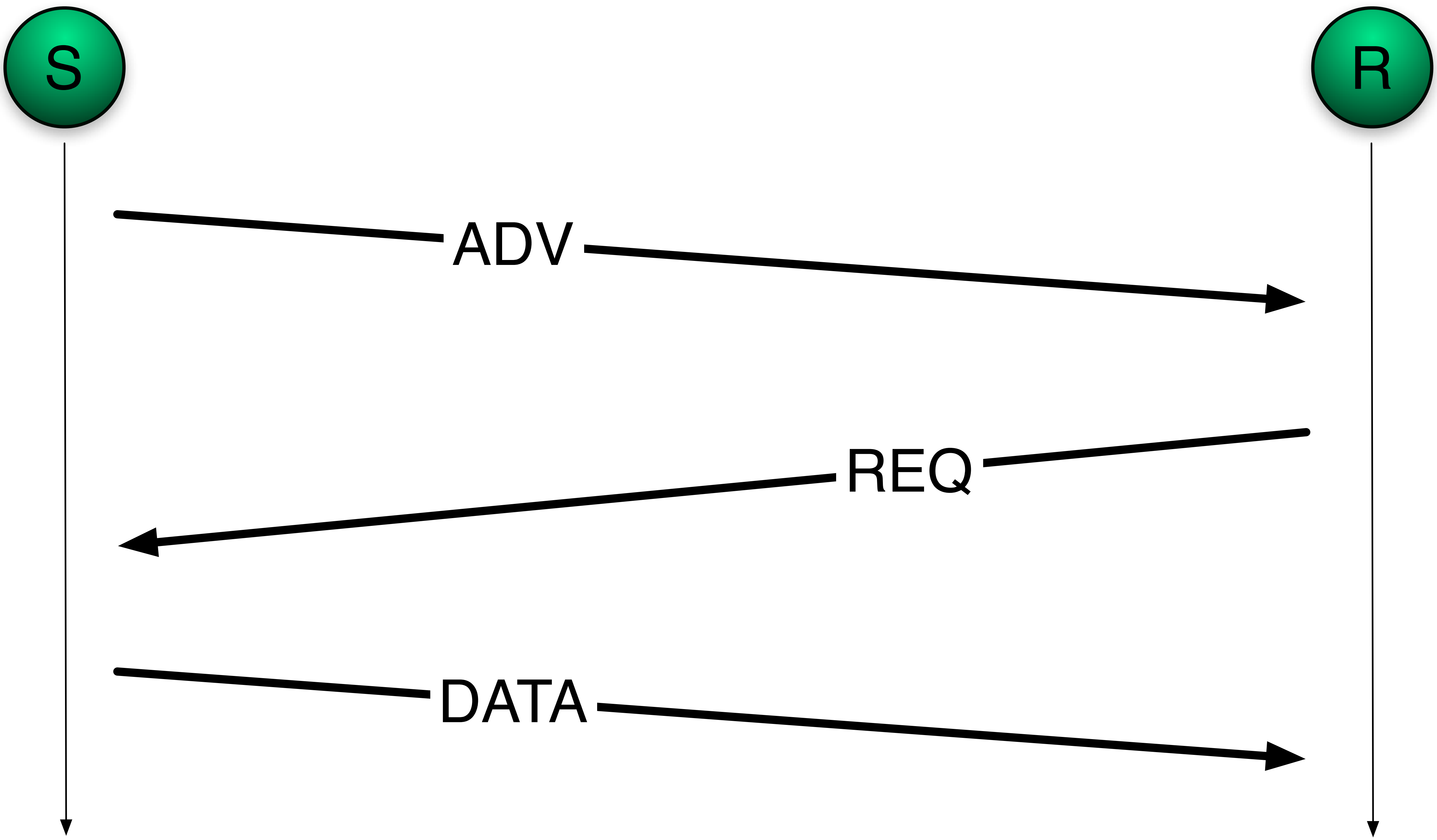
SPIN

- **ADV**: *advertize* that new data is available and described
- **REQ**: *request* to receive data
- **DATA**: actual *data*

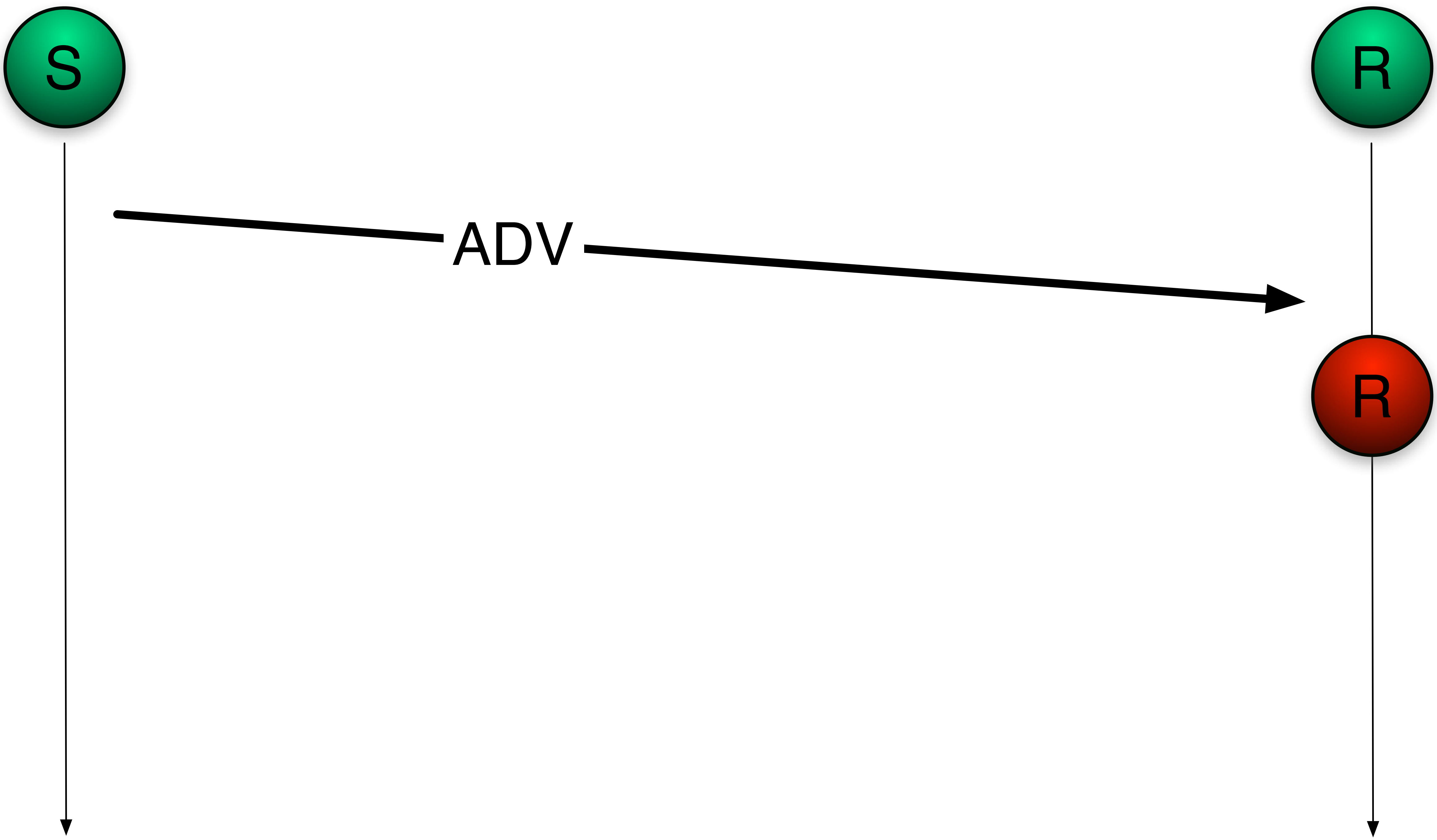
SPIN-PP



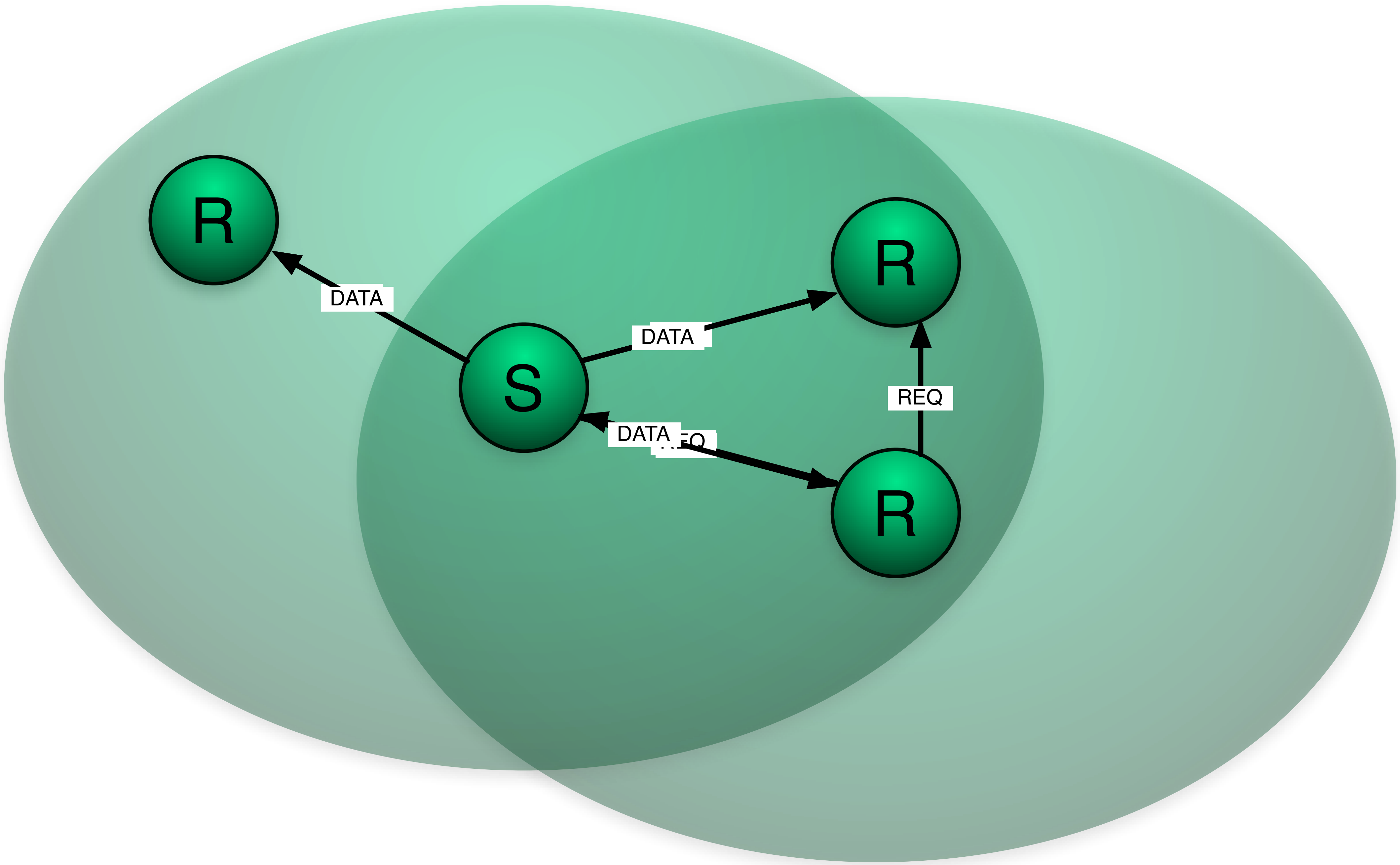
SPIN-EC



SPIN-EC



SPIN-BC, SPIN-RL



Directed Diffusion

- Destination-initiated (sink) reactive routing technique
- Data is named by an attribute-value pair
- Sensing tasks are initiated in order to match events *interests*
- All nodes maintain interest cache for each requested interest

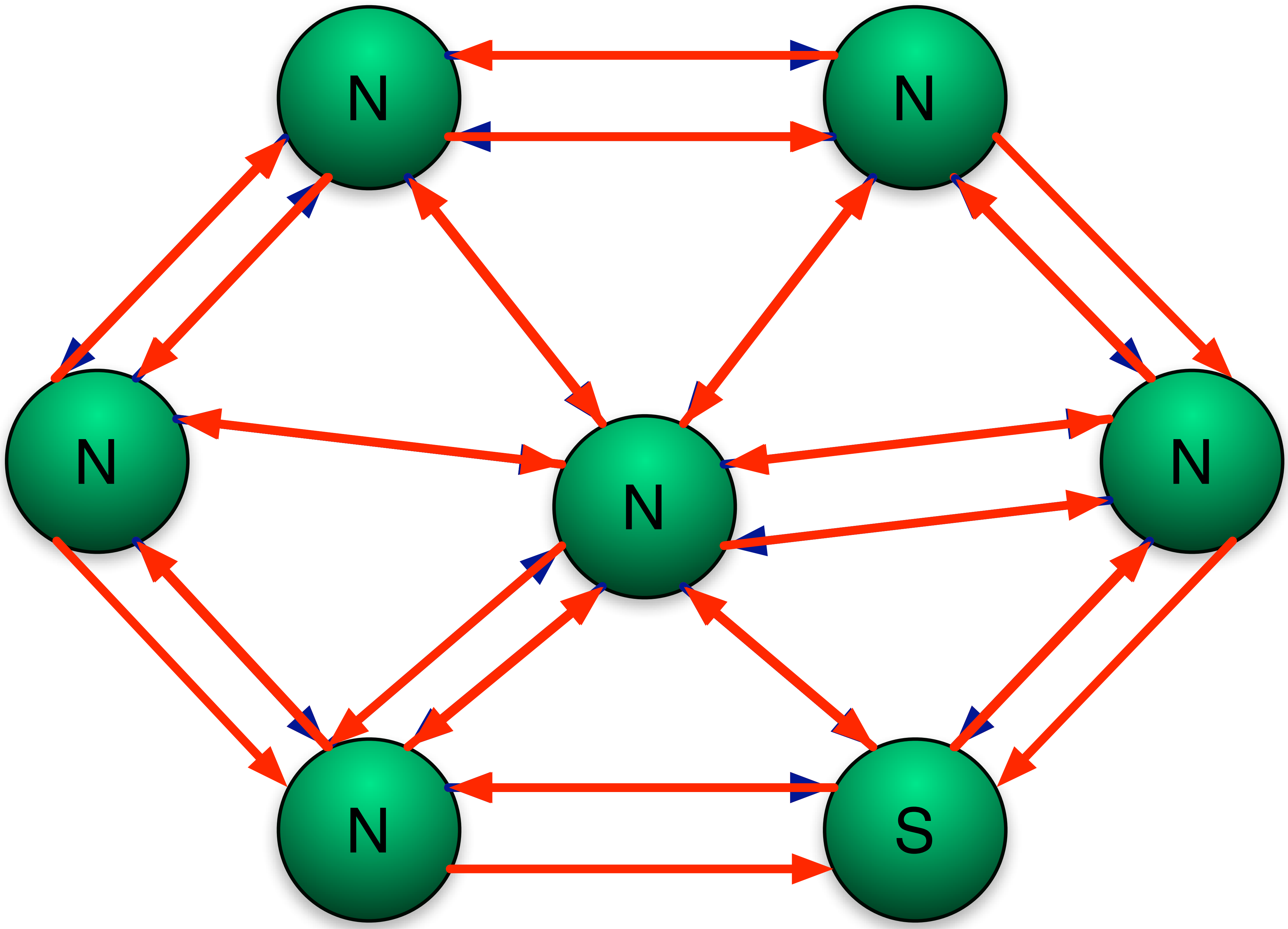
Interests

item name	value
type	four-legged animal
interval	20 ms
duration	10 s
rect	[-100, 100, 200, 400]

Returned Data

item name	value
type	four-legged animal
coordinates	[125, 220]
intensity	0.6
confidence	0.85
timestamp	01:20:40

Directed Diffusion



Interest Cache

- Periodically purged
- No information about sink
- Gradient table
  - *rate per neighbor*
  - *timestamp*
  - *expiration*

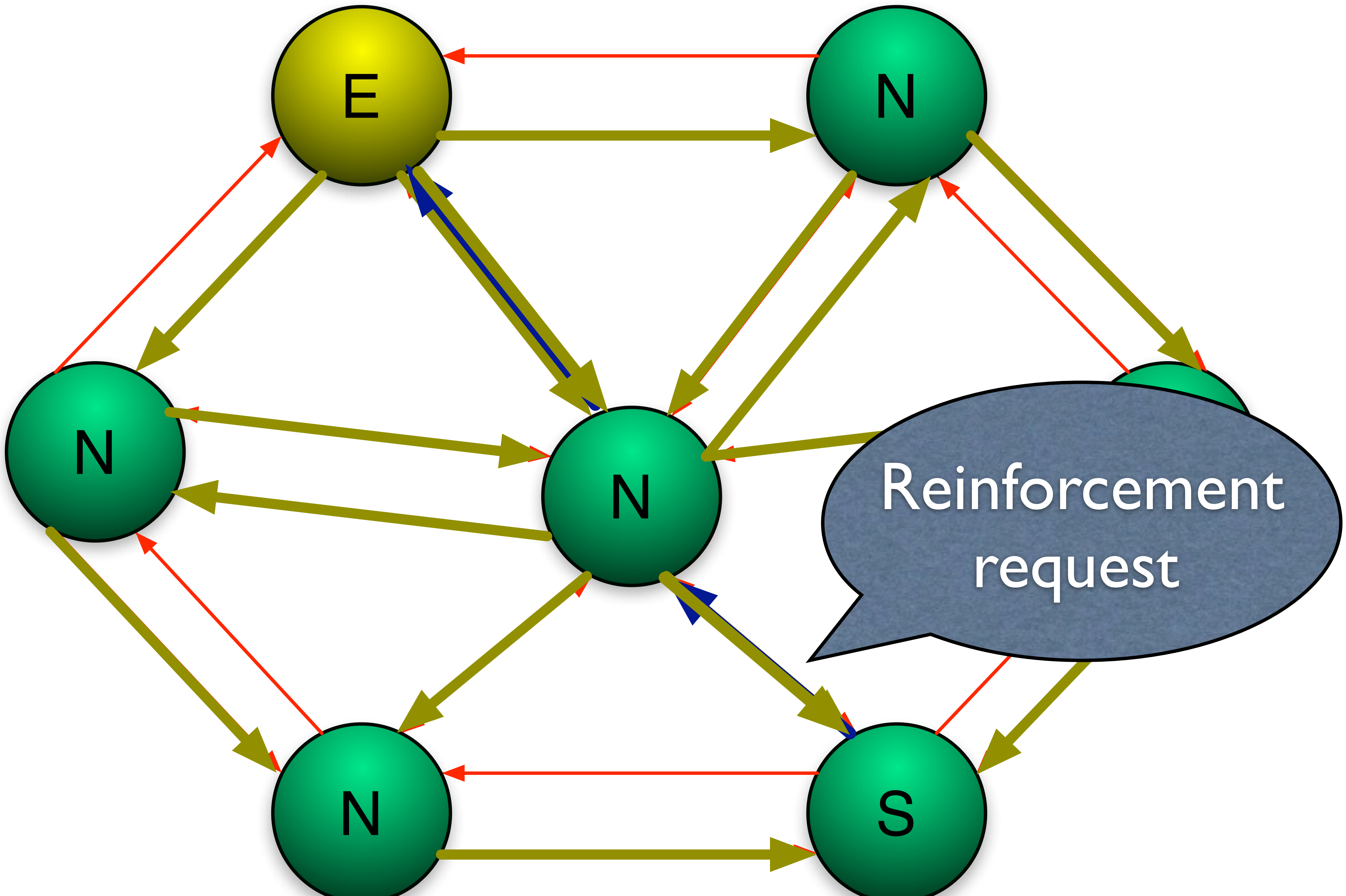
Interest Forwarding

- When new interest/task is received, add to cache
- Simplest policy: rebroadcast interest
- No way of distinguishing new interests from repeated ones
- Set up (very low rate) gradients between all neighbors

Message propagation

- A node matching an interest generates replies at desired rate
- When receiving a reply, lookup interest cache
- Forward along given route(s) if found, drop otherwise
- Loop prevention

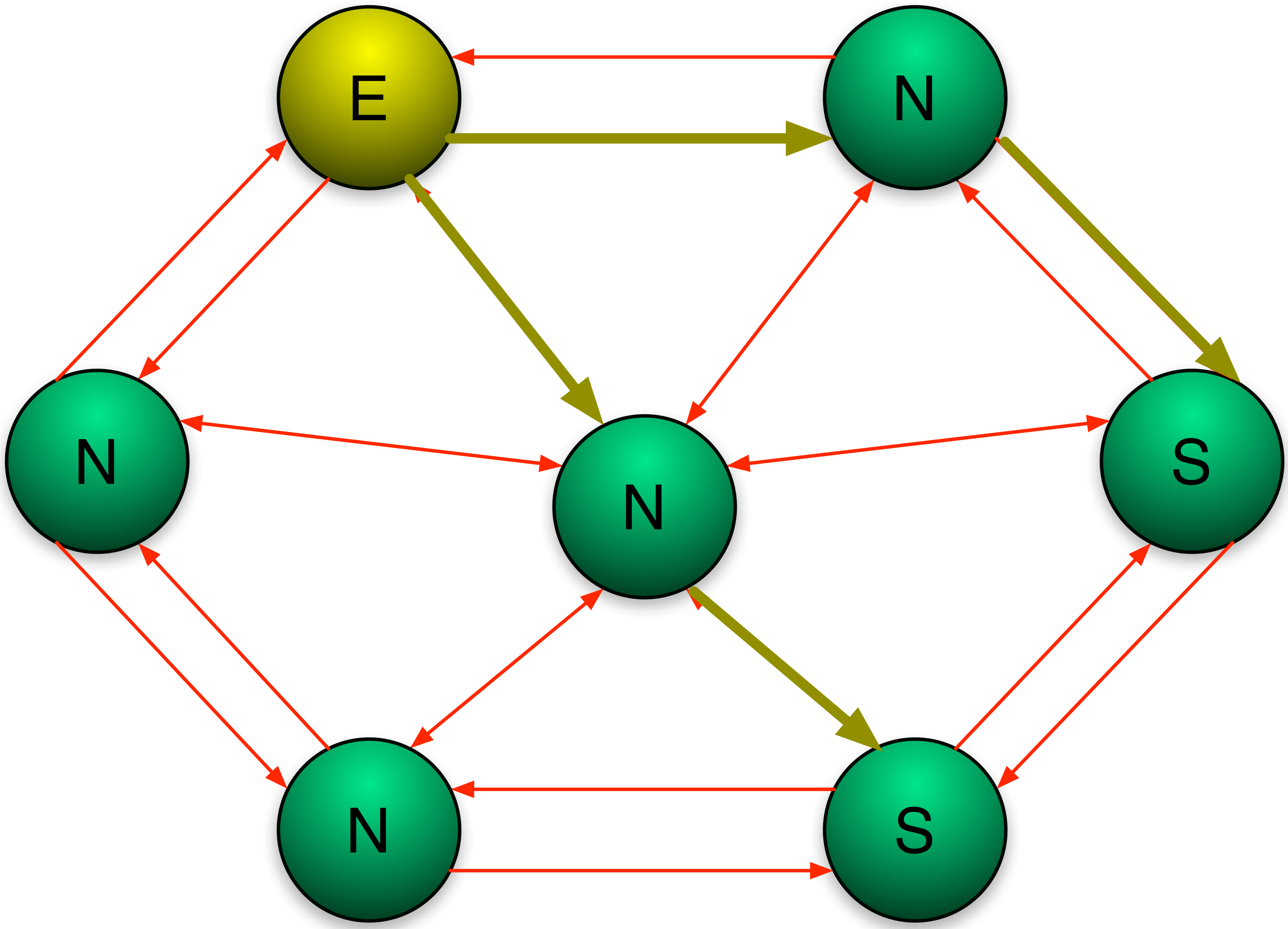
Directed Diffusion



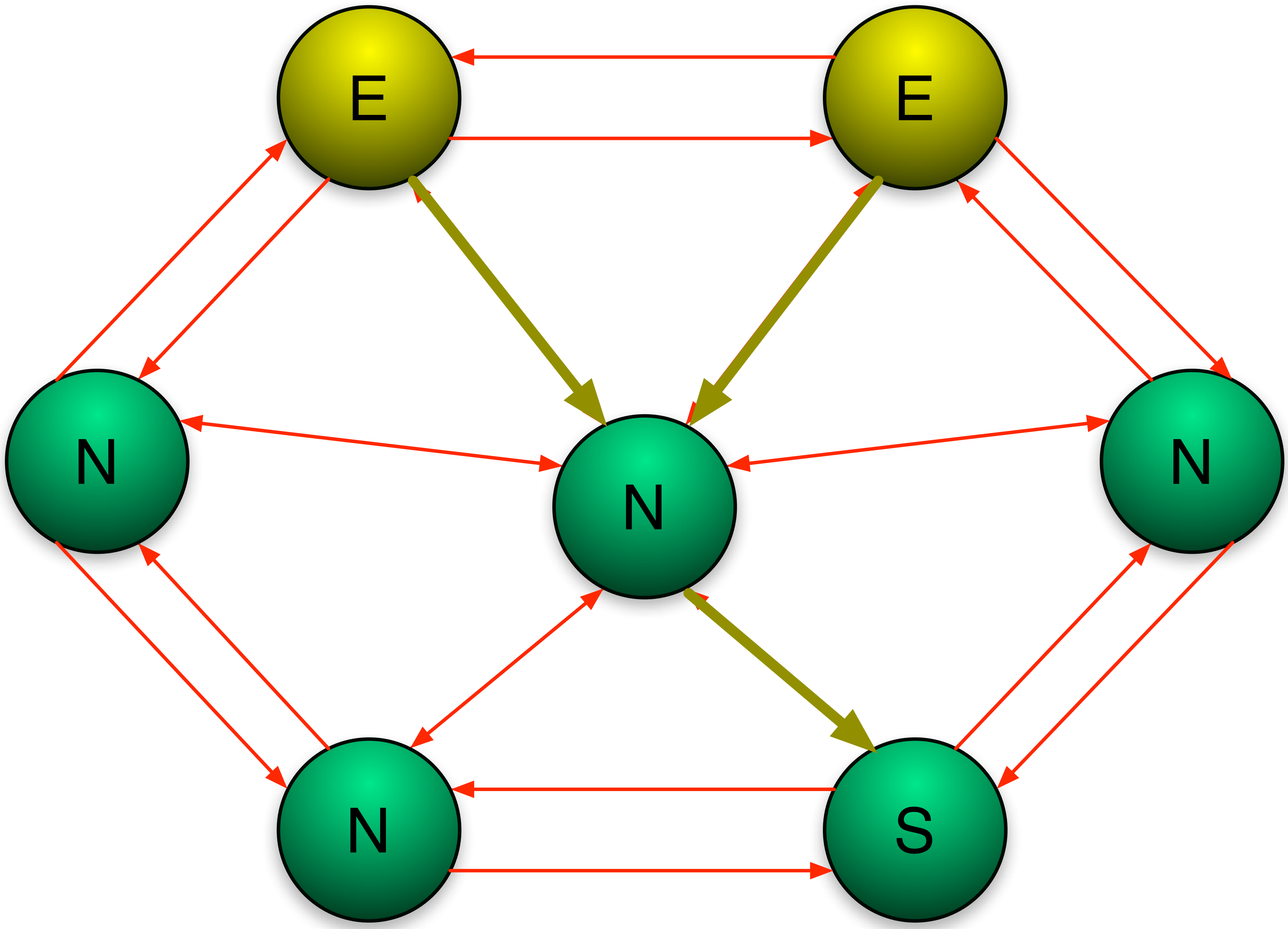
Reinforcement

- Sink can reissue the same request with a higher rate
- “Draw down” higher quality data from a particular neighbor
- Other nodes react when receiving
- “Outflow” increased, must reinforce another node to increase “inflow”

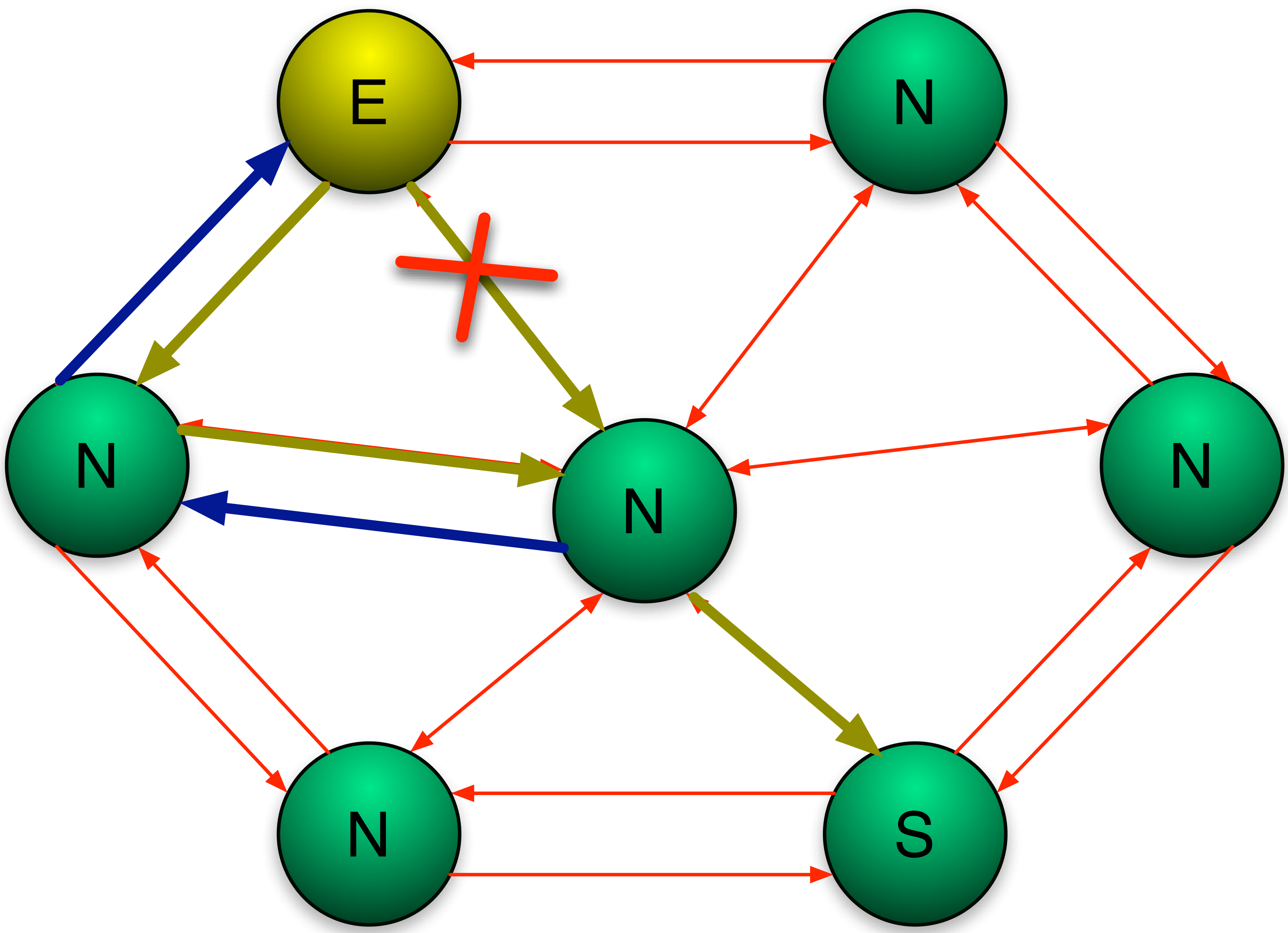
Directed Diffusion



Directed Diffusion



Directed Diffusion



Directed Diffusion

- **Local algorithm policies**
  - *Propagating interests*
    - flood, cache information, GPS
  - *Setting up gradients*
    - first heard neighbor, highest energy neighbor

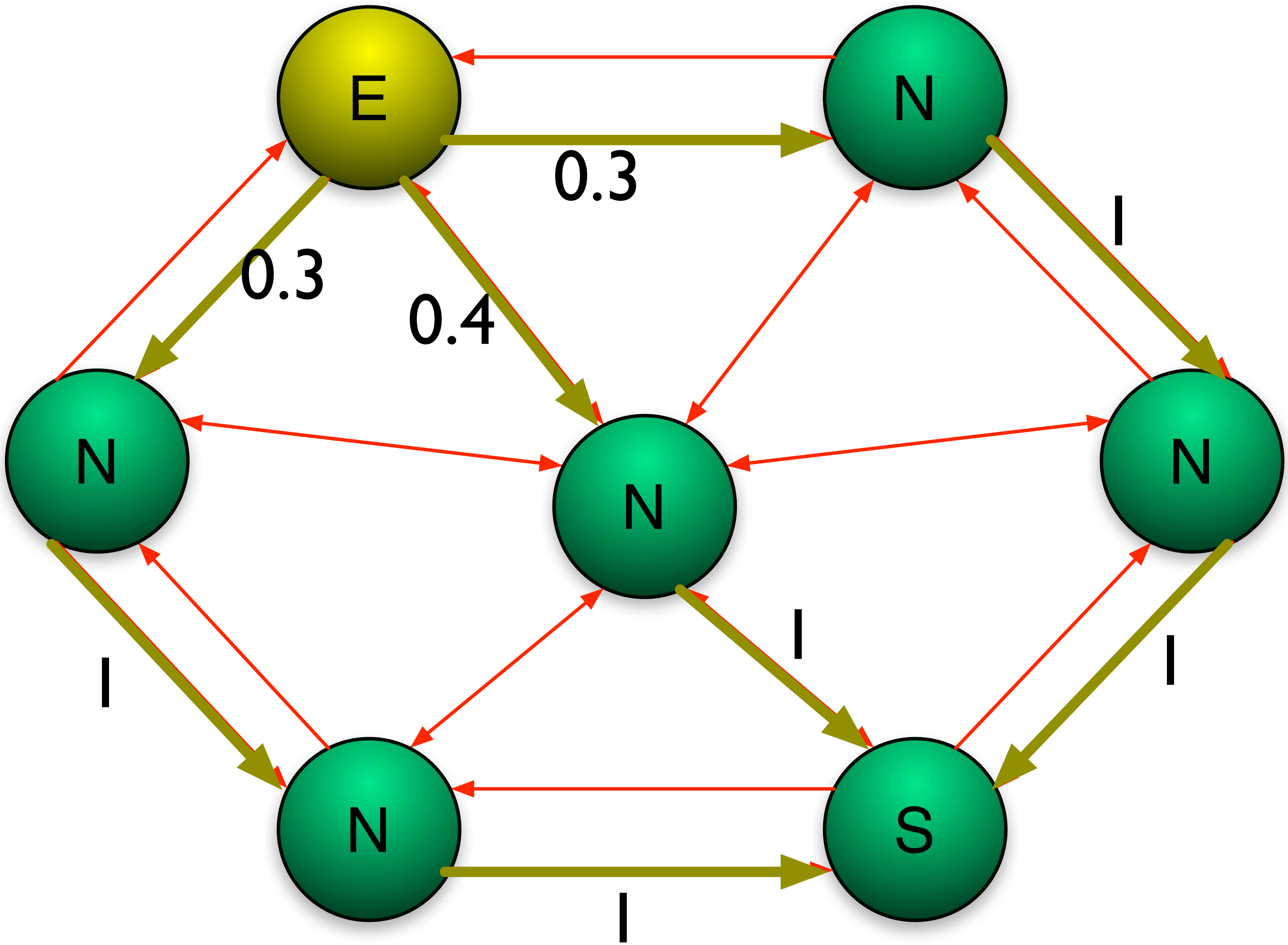
Directed Diffusion

- **Local algorithm policies**
  - *Data transmission*
    - single path, striped multi-path, multiple sources, etc.
  - *Reinforcement*
    - observer losses, resources levels, etc.

Energy Aware Routing

- Similar to Directed Diffusion
  - destination initiated
  - initial flooding to discover routes
  - several sub-optimal paths can be used (with a probabilistic distribution)

Energy Aware Routing



Medium Access Control

Classical Approaches

One frequency available

- **FDMA**: Frequency division multiple access

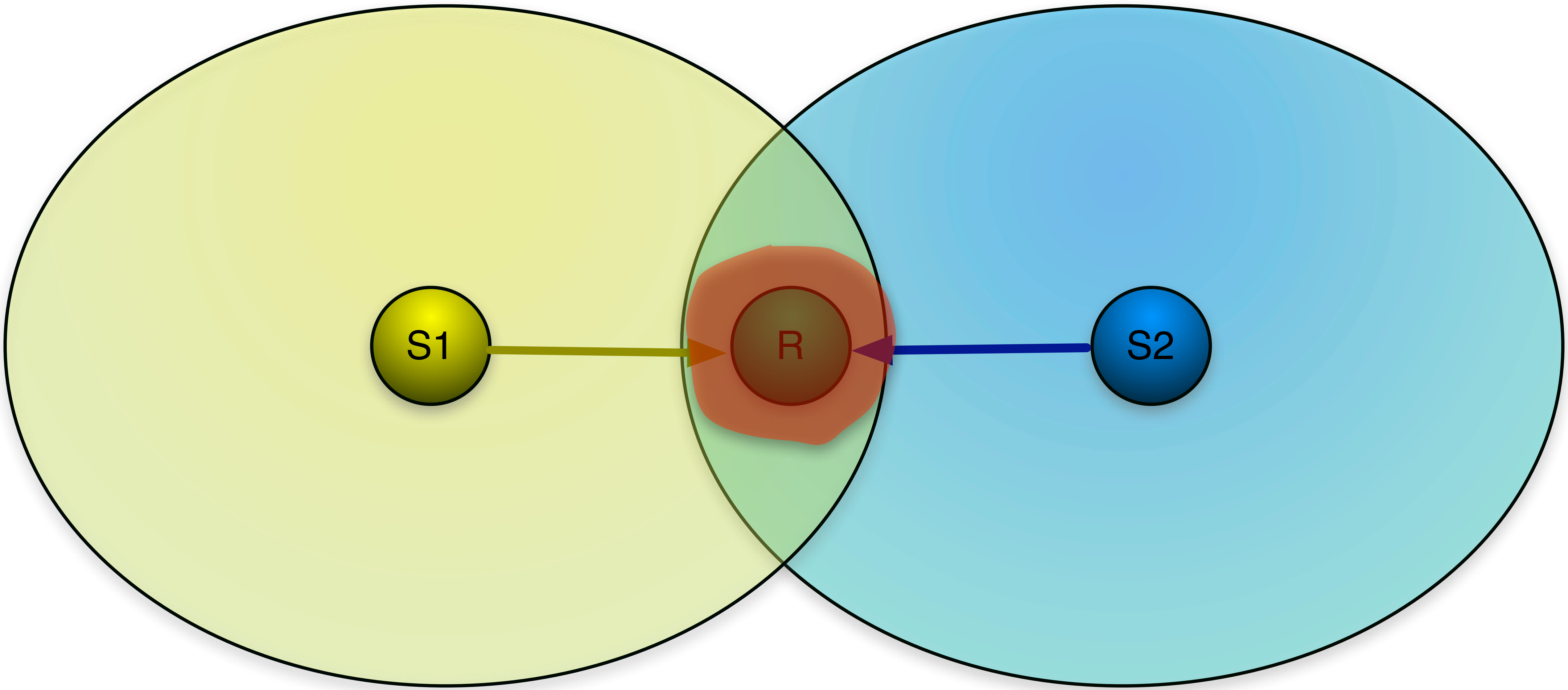
One code available

- **TDMA**: T
- **CDMA**: Code division multiple access

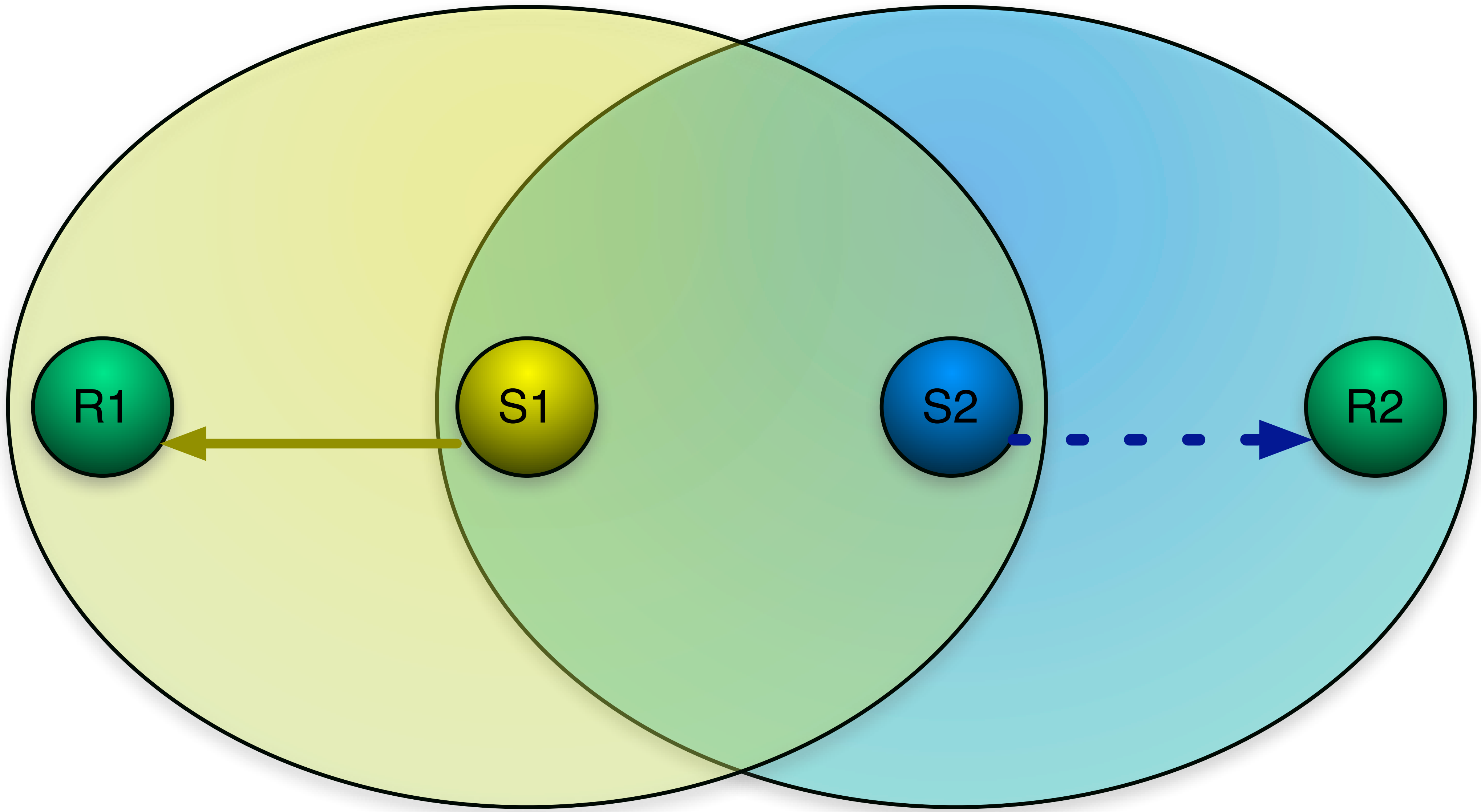
Can't listen while transmitting

- **CSMA**:
  - **CD**: Collision detection
  - **CA**: Collision avoidance

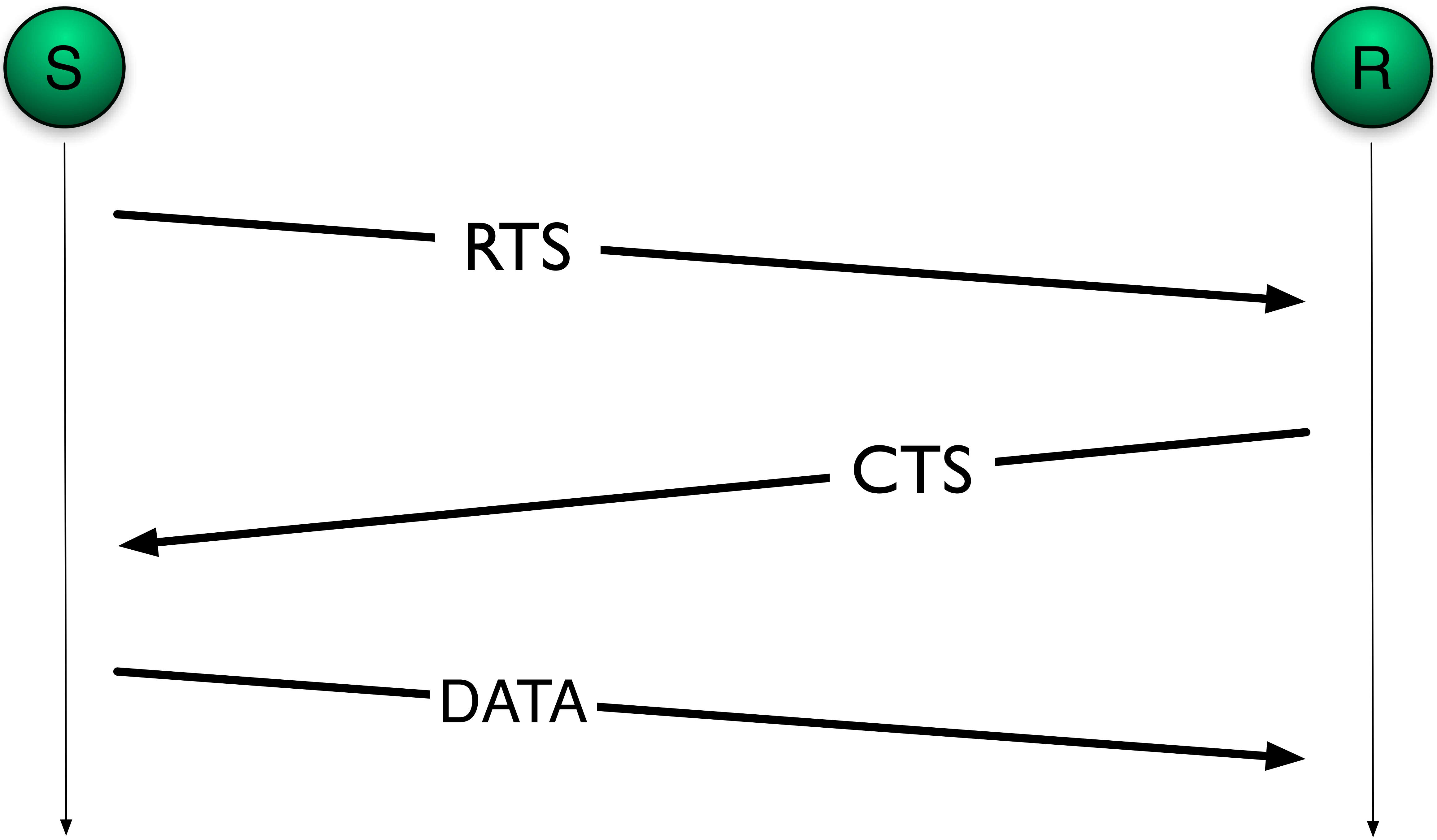
Hidden Terminal problem



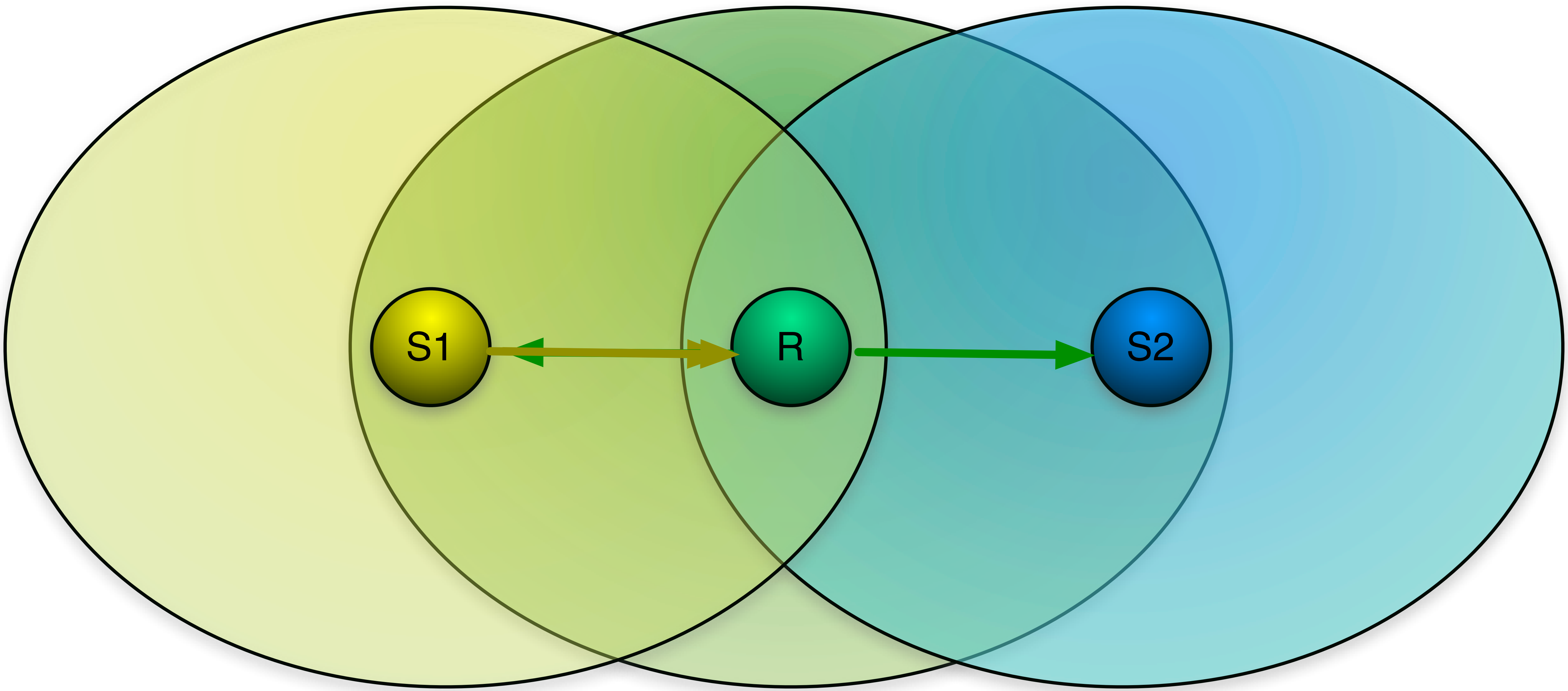
Exposed Terminal Problem



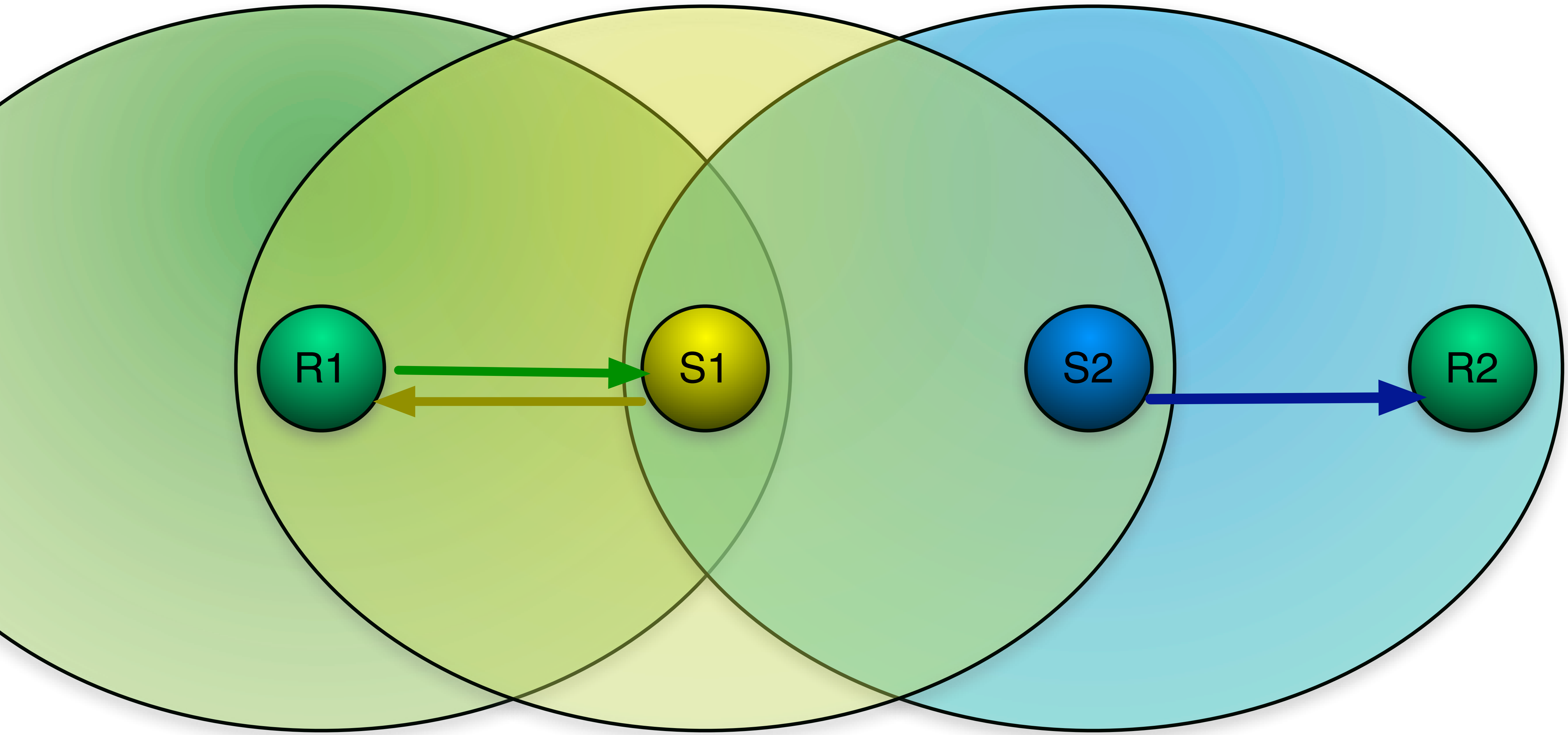
IEEE 802.11 RTS/CTS



IEEE 802.11 RTS/CTS



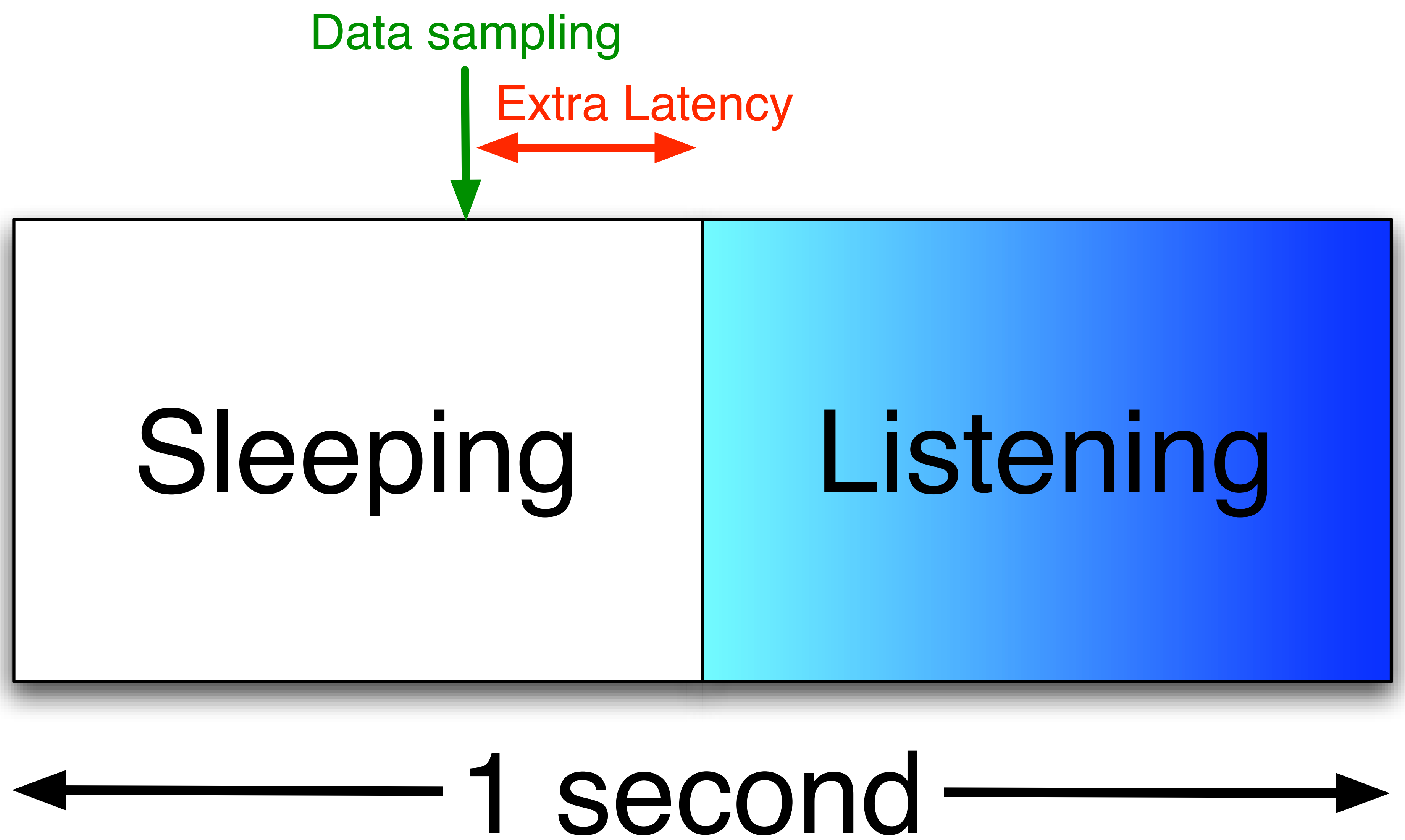
IEEE 802.11 RTS/CTS



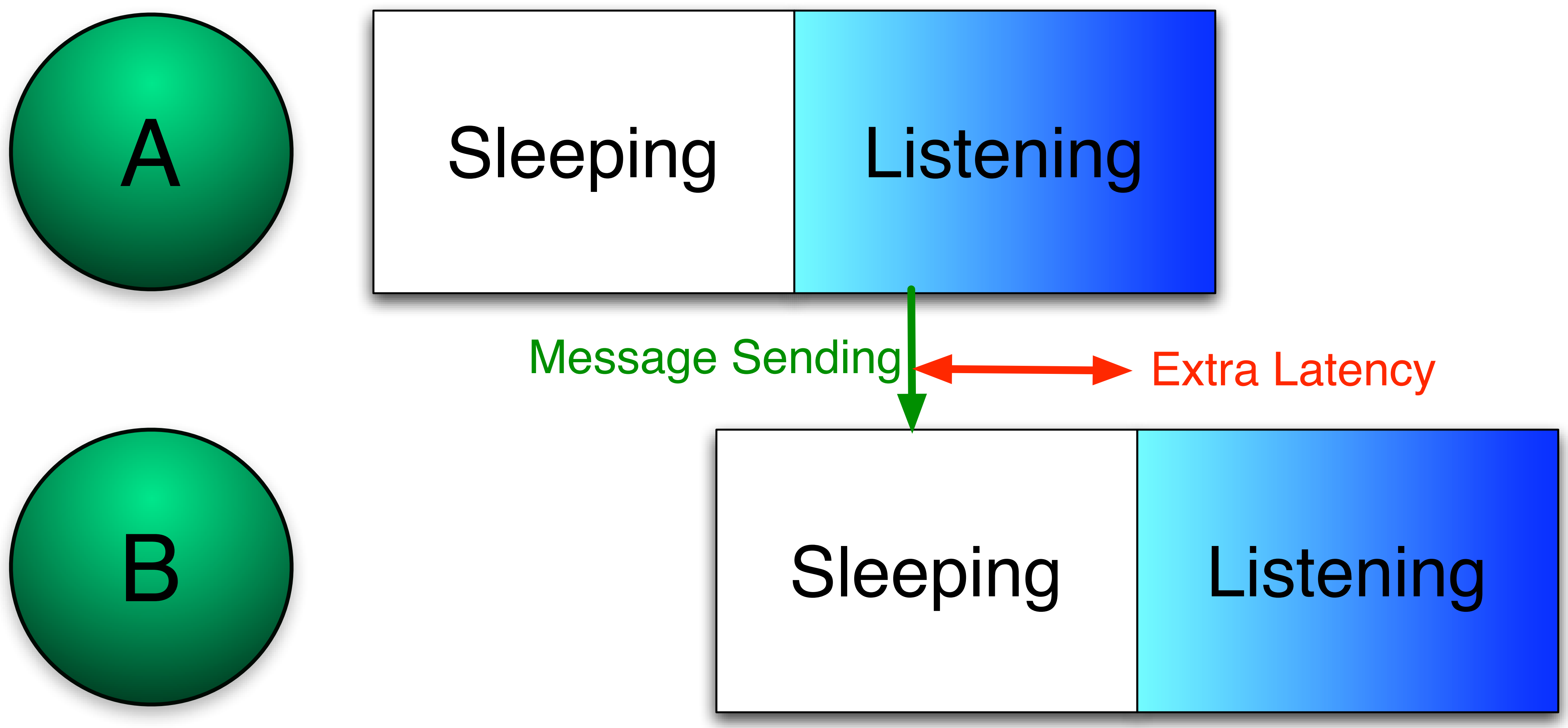
Duty Cycling

- **Reduces idle listening time**
  - Sensors switch between sleep and active mode
- **Suits low traffic networks**
  - If data rate is very low, it is not necessary to keep sensors listening all the time
  - Energy can be saved by turning off sensors

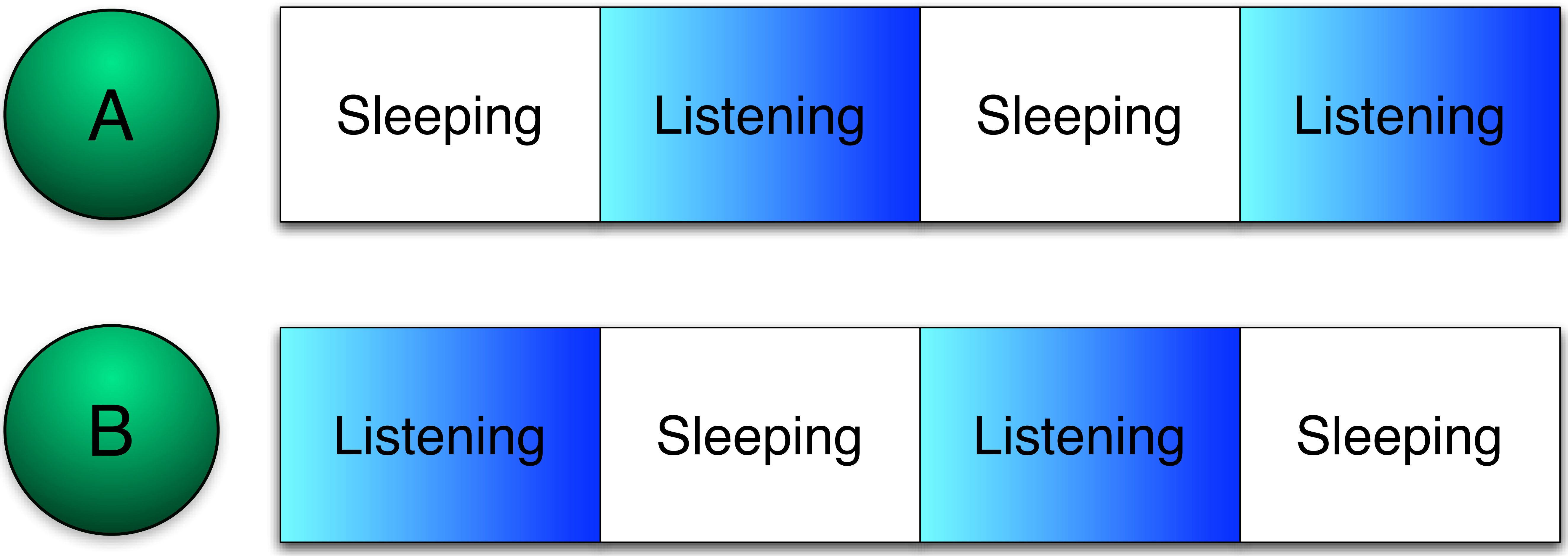
Duty Cycling



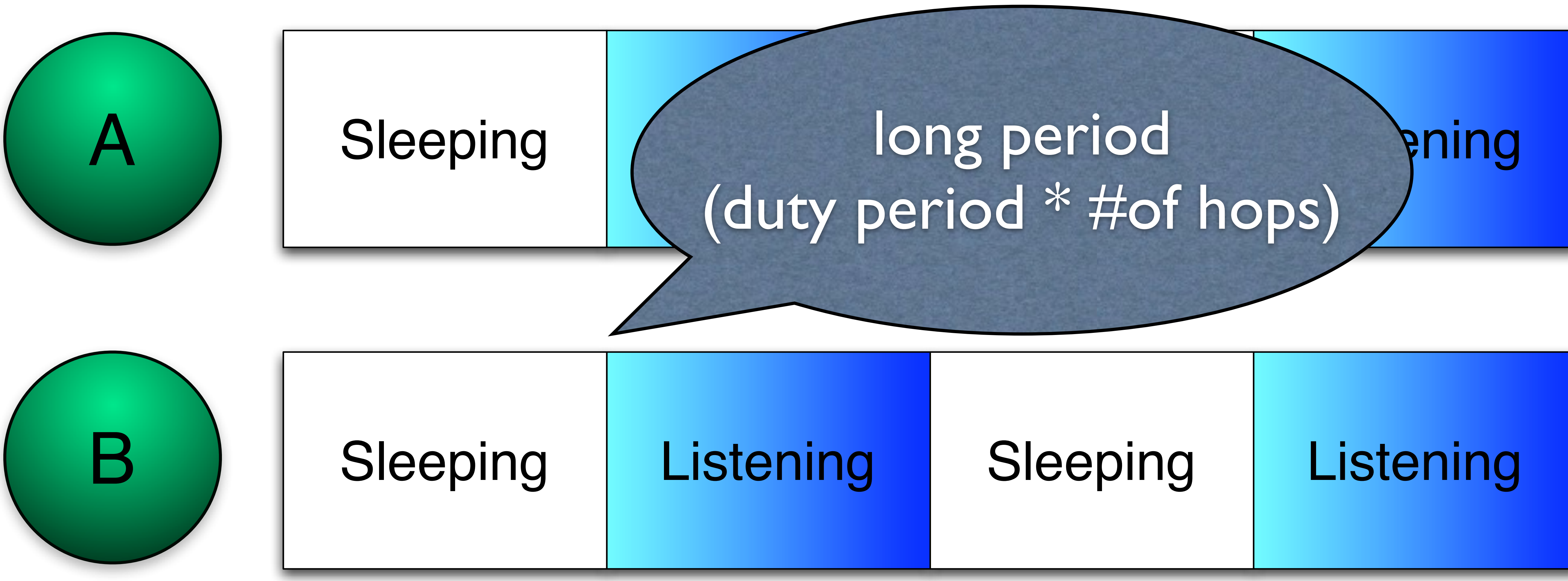
Duty Cycling



Duty Cycling

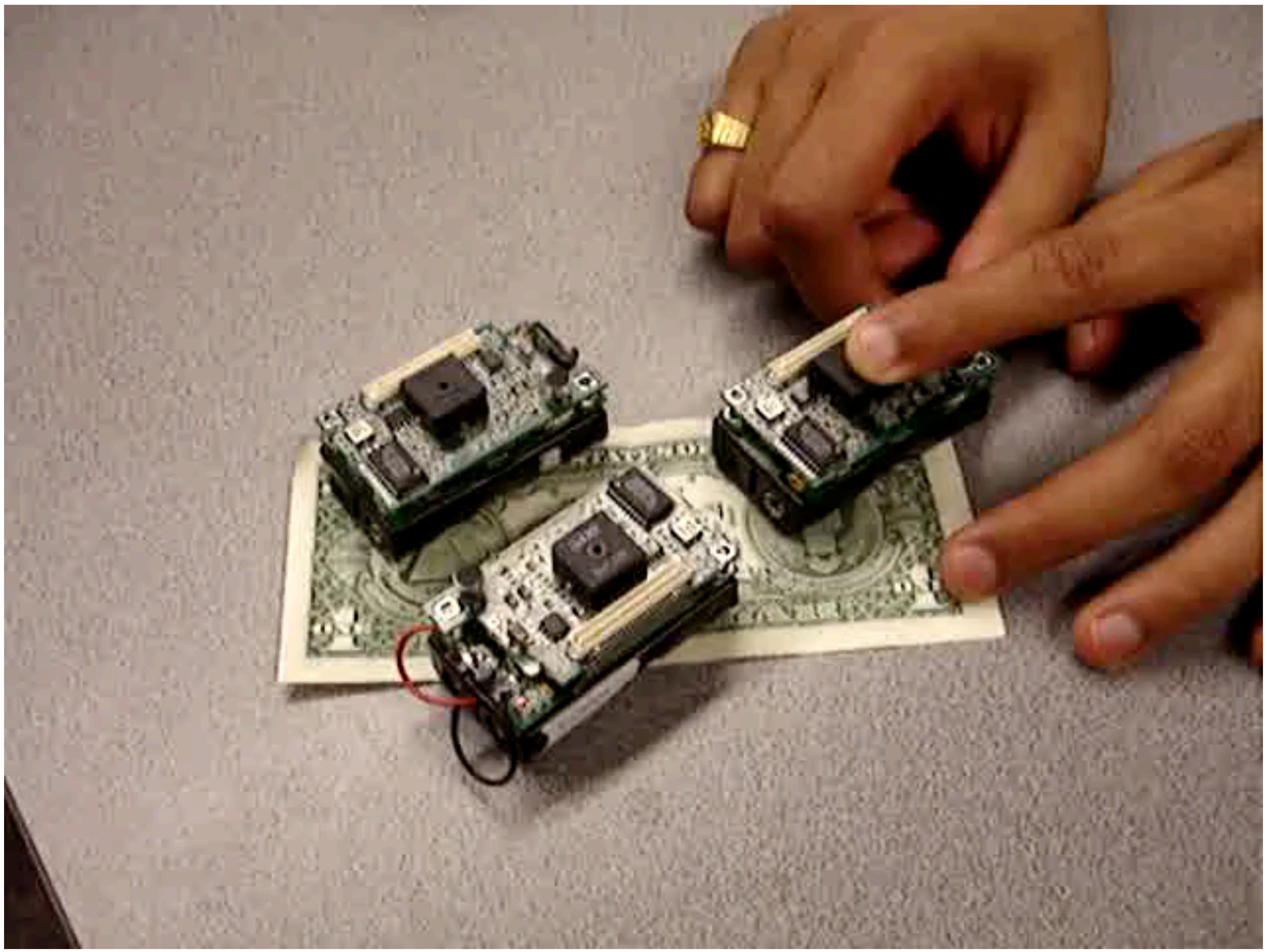


Duty Cycling



Time Synchronization

Time Synchronization



Time synchronization

- **Definition**: providing a common time scale for local clocks of nodes in the network
  - Stamp event, duration between events, order events
  - No global clock of shared memory

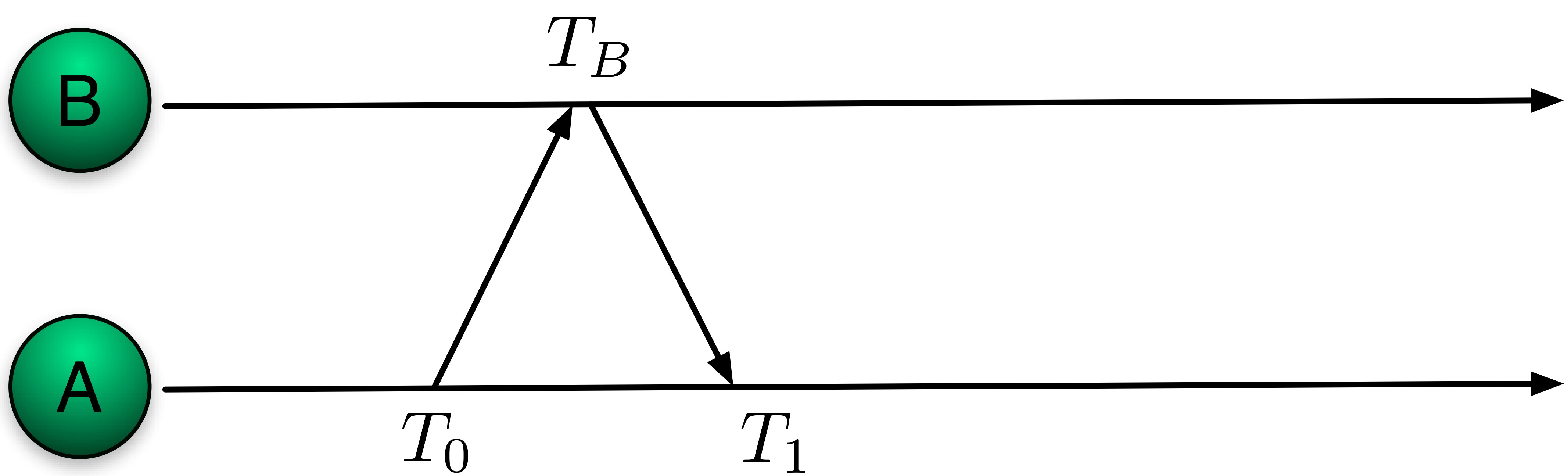
Time Synchronization

$$C_p(t) = a_p t + d_p$$

$$a_p : \text{clock frequency}$$

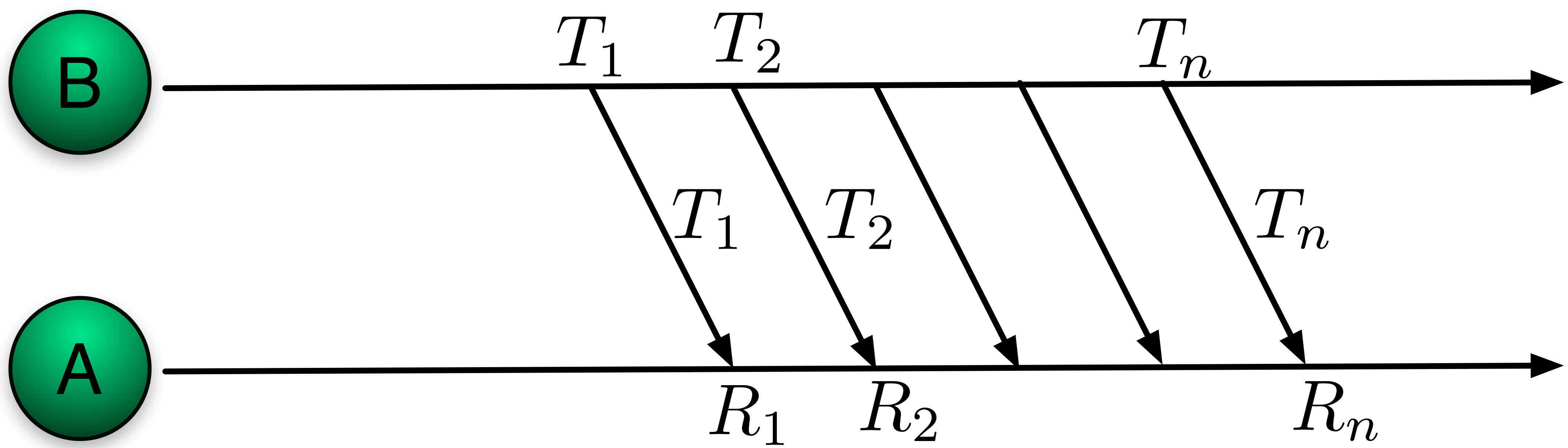
$$d_p : \text{offset}$$

Remote Clock Reading



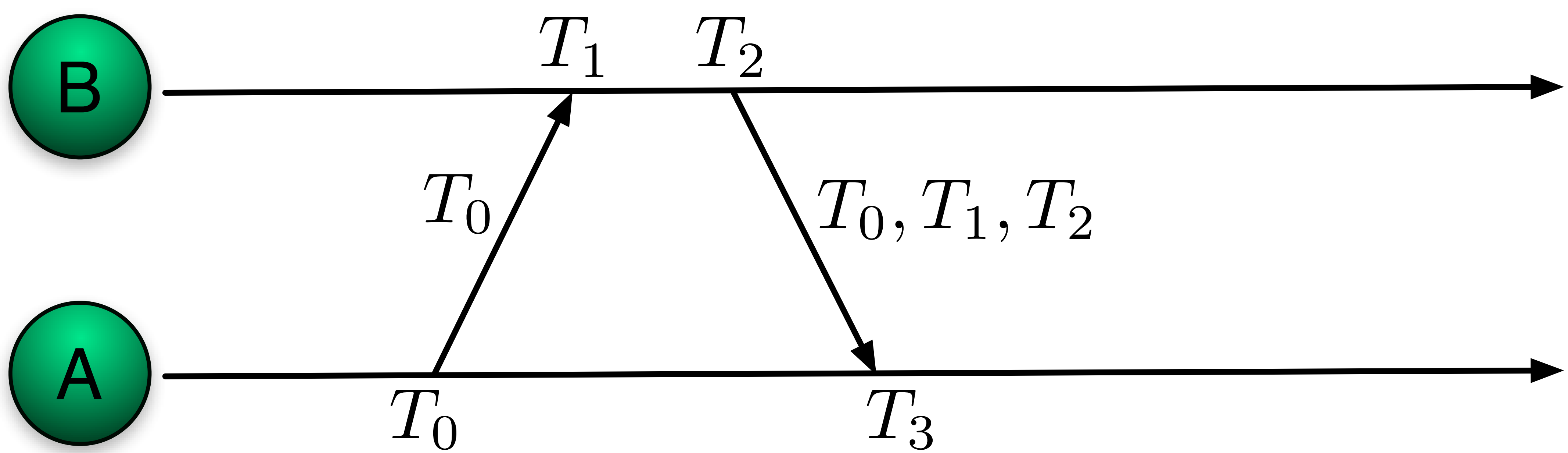
$$T_B(T_1) = T_B + \frac{T_1 - T_0}{2}$$

Time Transmission



$$T_B(R_n) = R_n - \left(\frac{1}{n} \sum_{i=1}^{n} R_i - \frac{1}{n} \sum_{i=1}^{n} T_i \right) + d$$

Offset Delay Estimation

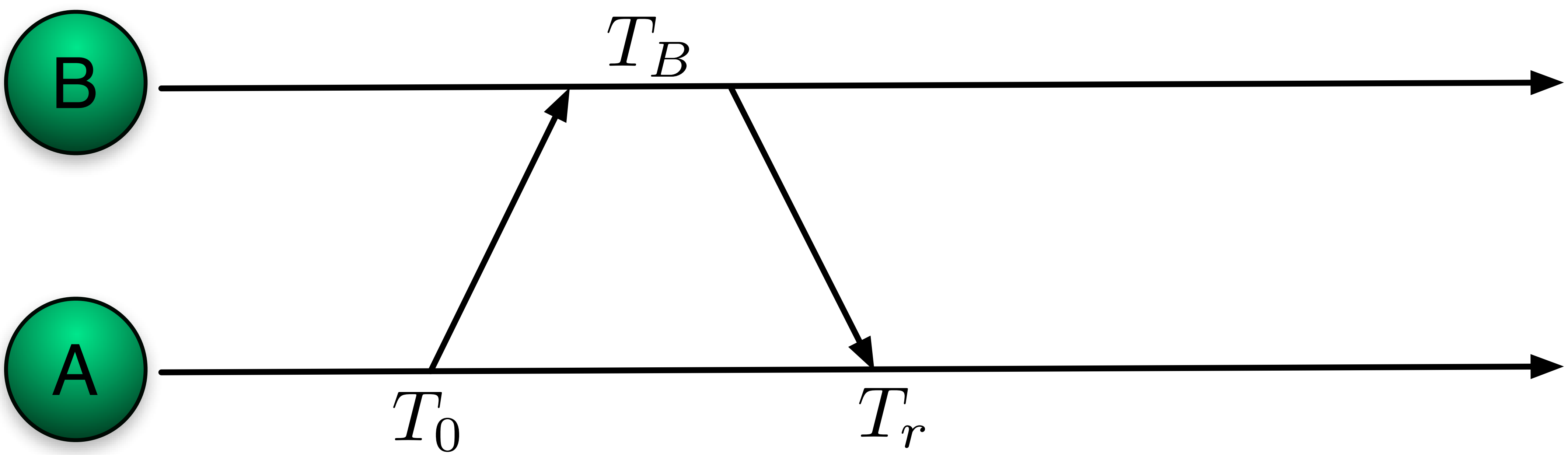


$$\Delta = \frac{(T_1 - T_0) - (T_3 - T_2)}{2} \qquad d = \frac{(T_1 - T_0) + (T_3 - T_2)}{2}$$

Drift

Offset

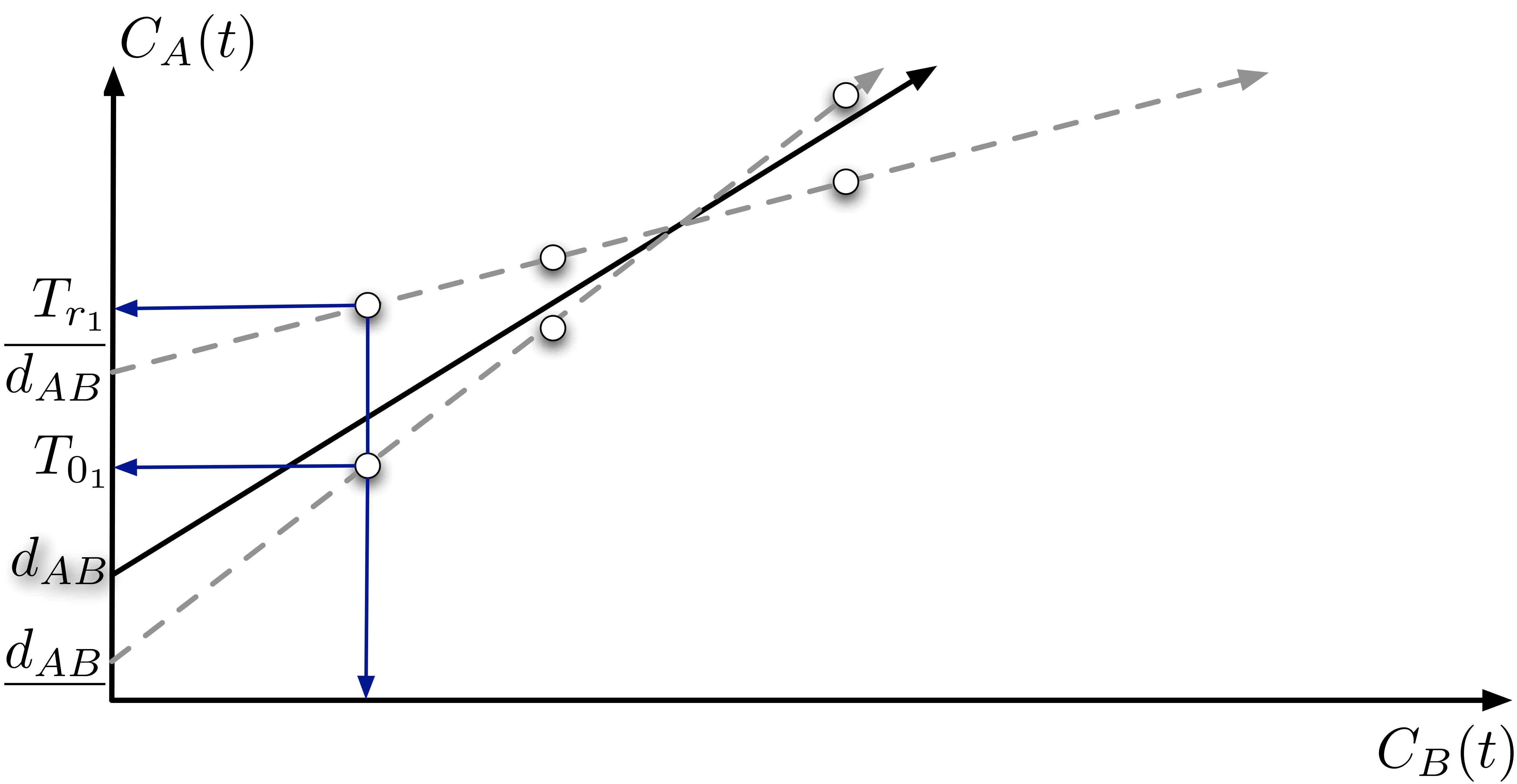
Set Valued Estimation



$T_0 < a_{AB}T_B + d_{AB}$ $\qquad$ $T_r > a_{AB}T_B + d_{AB}$

$$C_A(t) = a_{AB}C_B(t) + d_{AB}$$

Set Valued Estimation



Conclusion

Sensor Networks

- Driven by applications
- Connexion between Computer Science and Biology, Environment, Rescue, etc.
- Hard problems yet to be solved