

# Service Discovery in Sensor Networks: An Overview

Ramiro Liscano

School of Information Technology and  
Engineering

University of Ottawa, Ottawa, CANADA

<http://www.site.uottawa.ca/~rliscano>

## Overview

- ◆ Sensor Network Models
  - WEB-based Sensor Networks
  - Peer to Peer
  - Wireless Ad Hoc Sensor Networks
- ◆ Service Discovery Basics
- ◆ WEB Sensor Networks
  - Sensor Web Project
  - SensorML
- ◆ Service Discovery in Peer to Peer Networks
  - JXTA
- ◆ Wireless Sensor Networks and Service Discovery
- ◆ Summary

## Some Sensor Network Types

## Types of Sensor Networks

### ◆ WEB-based

- Sensors that primarily rely on the existing Internet protocols for communication.
- Can leverage Web service interfaces like SOAP & WSDL.
- In most circumstances require a gateway between a group of sensors or sensor to the Internet.

### ◆ Peer to Peer Networks

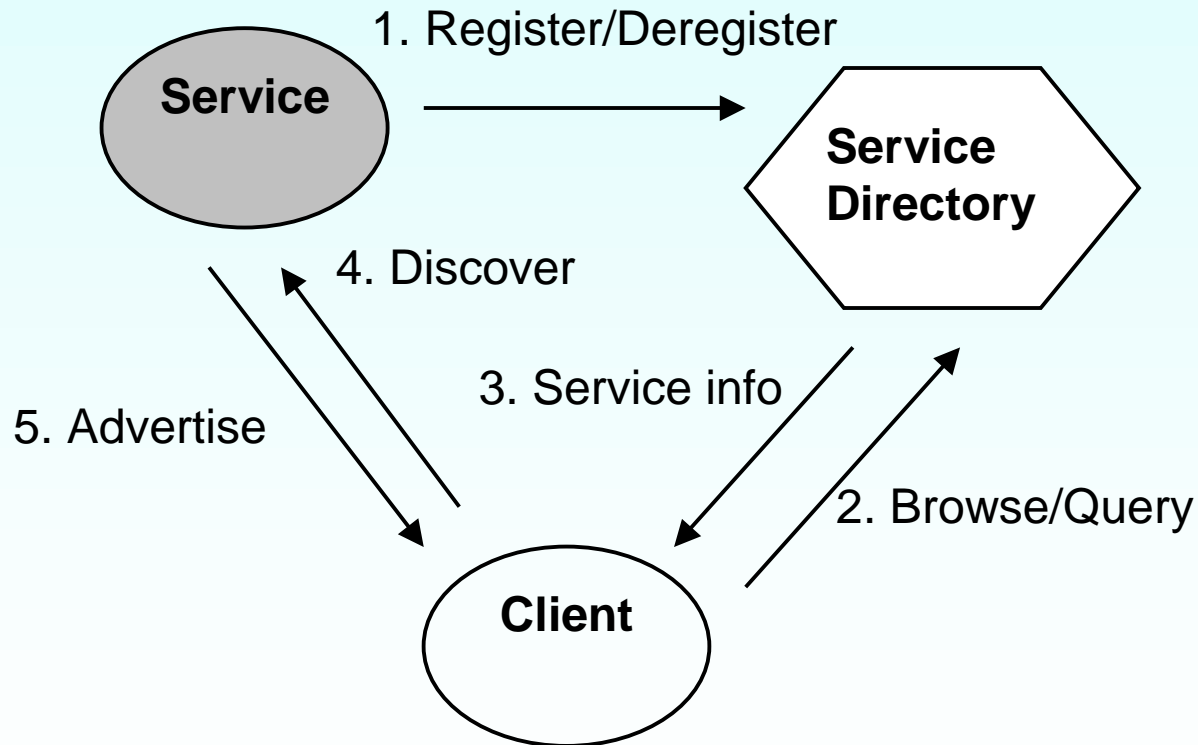
- Focused on efficient routing and service discovery algorithms for gathering sensor information.

### ◆ Wireless Ad hoc Networks

- Concentrate mostly on routing algorithms that need to conserve power.
- RF coverage is also a driving factor in design.
- Routing is based on sensor requests rather than node address.

# Service Discovery Basics

## Service Discovery: Entity Interactions



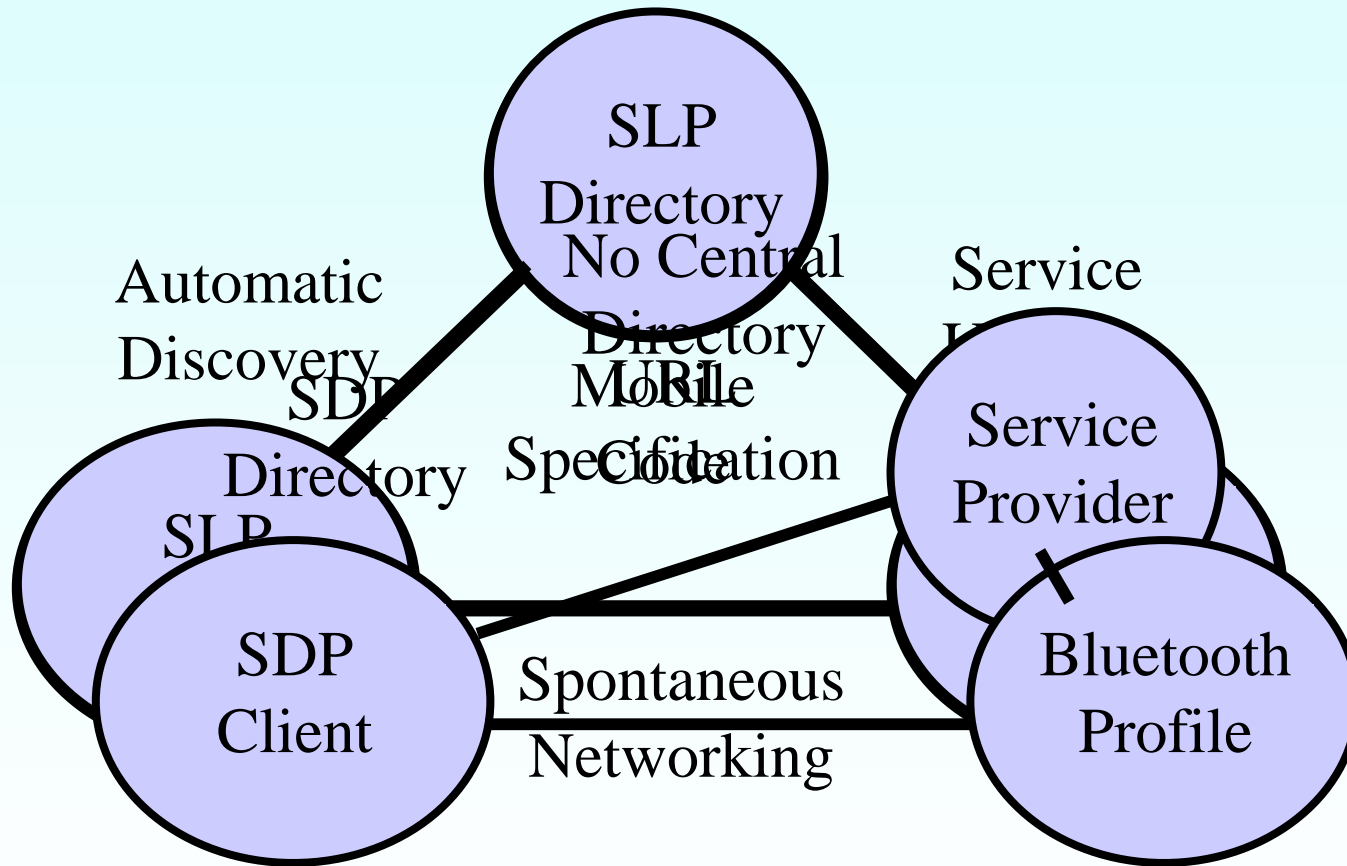
# Components in a Service Discovery Protocol

- ◆ Service Description
  - Defines a service profile or template.
- ◆ Service Query
  - Defines a language and mechanism by which services can be discovered.
- ◆ Service Acquisition and Use
  - Defines a mechanism by which the service is delivered.
- ◆ Framework
  - Defines the components and interaction among those components in order to discover, acquire, and use the service.
- ◆ Leasing (not really a key component)
  - Lifetime of a service

## Common Service Discovery Protocols for the Enterprise

- ◆ Sun's Jini, a java-based distributed model.
- ◆ IETF Service Location Protocol (SLP).
- ◆ Microsoft's UPnP.
- ◆ Bluetooth's Service Discovery Protocol (SDP).
- ◆ ...

## Common Service Discovery Framework



## Questions about Sensor Service Discovery

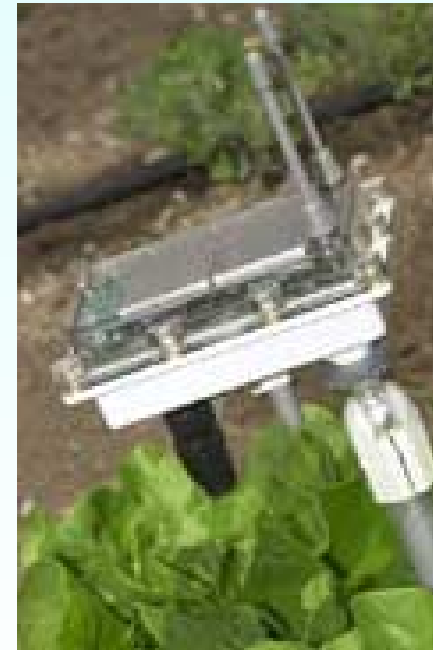
- ◆ How are sensor services described?
- ◆ How does a sink for sensory data request for sensory data services that are available in a sensor network?
  - How does the request for a service / data propagate from one point to another in the network?
- ◆ How is sensor data handled and processed within a network?

# The Web as a Model for Sensor Networks: Functional Sensor Models

## Example of Web-based sensor networks

### ◆ SensorWeb@GeoICT or JPL Sensor Web

- Focused on the development of a sensor network for geo-spatial applications.



<http://sensorwebs.jpl.nasa.gov/>

<http://www.opengis.org>

<http://vast.nsstc.uah.edu/SensorML/>

## Objectives set by OpenGIS Consortium

- ◆ **All sensors connected to the Web**
- ◆ **All sensors report position & observations**
- ◆ **Sensor modeling and encoding: SensorML**
- ◆ **Access observations through Sensor Collection Services**
- ◆ **Plan collections through Sensor Planning Services**
- ◆ **Access sensor-related metadata through Web Registry Services**
- ◆ **Messaging through Web Notification Services**

## Areas of Concentration

- ◆ **SensorML** – The information models and encodings for sensors
- ◆ **Observations & Measurements** – The information models and encodings for observations and measurements
- ◆ **Sensor Collection Service** - Fetches Observations for a sensor or sensor constellation
- ◆ **Sensor Planning Service** – Determines collection feasibility plans and processes collection requests for a sensor or sensor constellation
- ◆ **Web Notification Service** – Executes and manages message dialogue between client and Web service(s) for long duration processes (asynchronous)

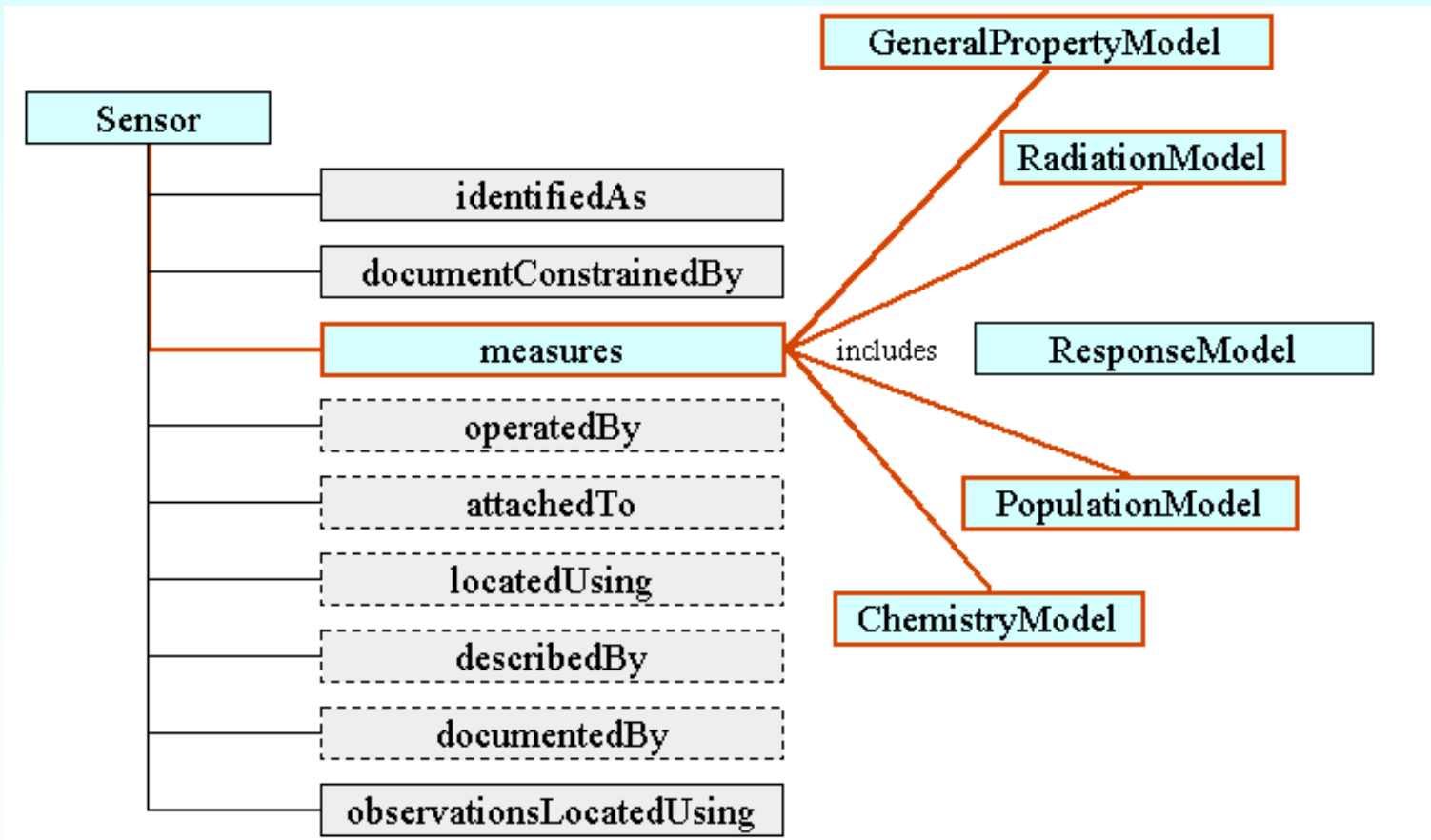
## SensorML – How are Services Described?

- ◆ **SensorML** is an information model and XML encodings for discovering, querying and controlling Web-resident sensors.
- ◆ SensorML provides a functional model of the sensor system, rather than a detailed description of its hardware.
- ◆ Originally developed at NASA Interuse Experiment (UAH, JPL: '93–'96) by Bill Taber of JPL.
  - Exploited by Mike Botts from the University of Alabama, Earth Science Systems for the Open GIS Consortium
  - <http://www.opengis.org> & <http://vast.nsstc.uah.edu/SensorML/>
- ◆ Designed for both dynamic and stationary platforms

## Capabilities of SensorML

- ◆ Provides general sensor information in support of **data discovery**
- ◆ Supports the **processing and analysis** of the sensor measurements
- ◆ Supports the **geolocation** of the measured data.
- ◆ Provides **performance characteristics** (e.g. accuracy, threshold, etc.)
- ◆ **Archives** fundamental properties and assumptions regarding the sensor

# SensorML Schema:



## SensorML Properties

- ◆ identifiedAs – Name of the sensor (shortname, longname, & sensor type)
  - Dictionary required for the sensor type.
- ◆ documentConstraintBy – Time period and authority for the calidity of the data.
  - Time and access constraints.
- ◆ Measures – Defines what type of property is being measured.
  - Based on measurand type.
- ◆ attachedTo – Described the type of platform the sensor is attached to.
  - Aircraft, land vehicle, satellite, etc.

## SensorML Properties

- ◆ hasCRS – Coordinate reference system.
- ◆ locatedUsing – Provides the basis for determining state of the sensor.
  - The objectState will maintain the required information for the object.
- ◆ operatedBy – Who has authority to operate sensor.
- ◆ describedBy & documentedBy – Provides meta-data and history of the sensor.

## XML Reply Example

```
<characterizedBy>
  <dynamicRange axis="http://www.opengis.net/observables#windSpeed">
    <low uom="http://www.opengis.net/units#mph">0.0</low>
    <high uom="http://www.opengis.net/units#mph">134.0</high>
  </dynamicRange>
  <threshold uom="http://www.opengis.net/units#mph"
    axis="http://www.opengis.net/observables#windSpeed">2.2</threshold>
  <operationalRange axis="http://www.opengis.net/observables#temperature">
    <low uom="http://www.opengis.net/units#celsius">-40.0</low>
    <high uom="http://www.opengis.net/units#celsius">40.0</high>
  </operationalRange>
  <survivableRange axis="http://www.opengis.net/observables#windSpeed">
    <low uom="http://www.opengis.net/units#mph">0.0</low>
    <high uom="http://www.opengis.net/units#mph">220.0</high>
  </survivableRange>
</characterizedBy>
```

## Remarks on SensorML

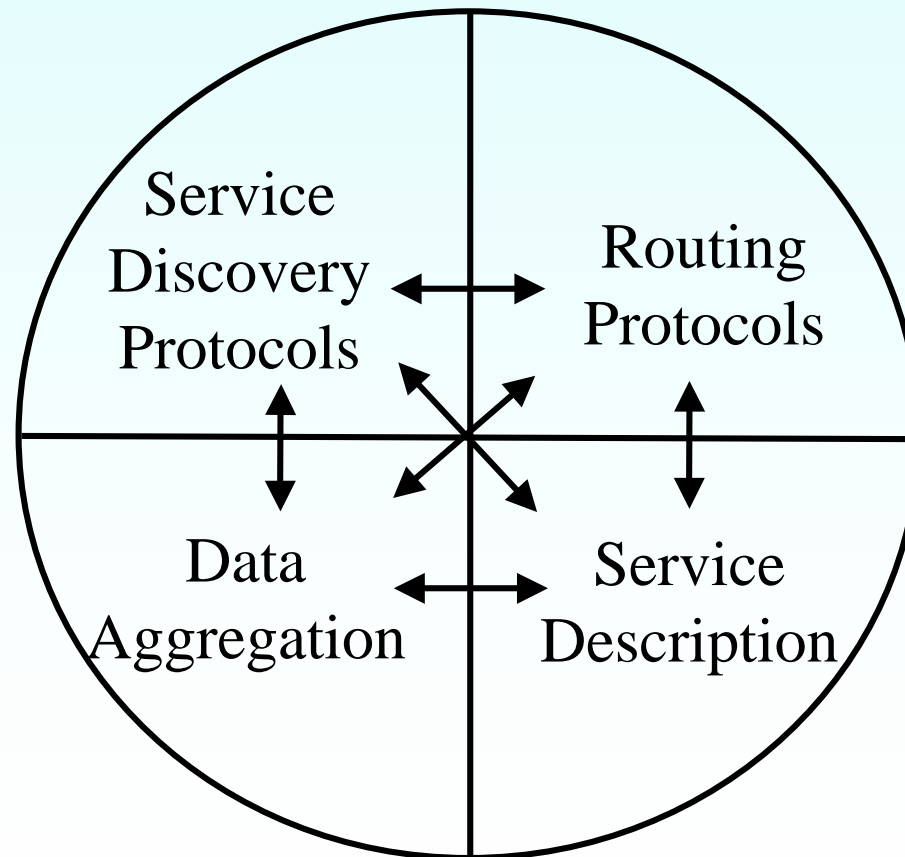
- ◆ Very comprehensive description of sensor functionalities for Geospatial applications.
- ◆ Not much if any documentation on the discovery of services.
- ◆ No concepts of the infrastructure requirements for service discovery. (i.e. use of distributed data-bases, name spaces, peer to peer).
  - How does a service requester know what services are available in a sensor network?

## How are the services propagated?

- ◆ Well this is not that clear.
  - Classical models based on sensor collection points and strong use of Web-based access technology is being proposed. (i.e. SOAP, UDDI,)
  - There appears to be no requirement for self-configuration. That is there is really no movement of nodes within the network.
- ◆ The Open-GIS Consortium is a good knowledge source for sensor modeling, but not for routing in ad hoc sensor networks.

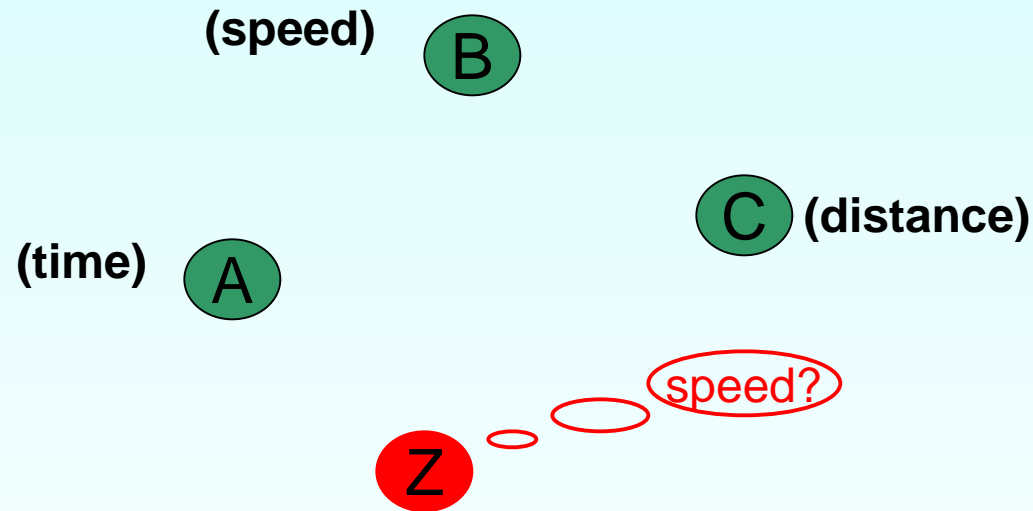
## Mobility and Sensor Networks

- ◆ What happens to service discovery when nodes are mobile or network is re-configurable?



# Small Tutorial on Service Discovery

# Service Discovery Concepts



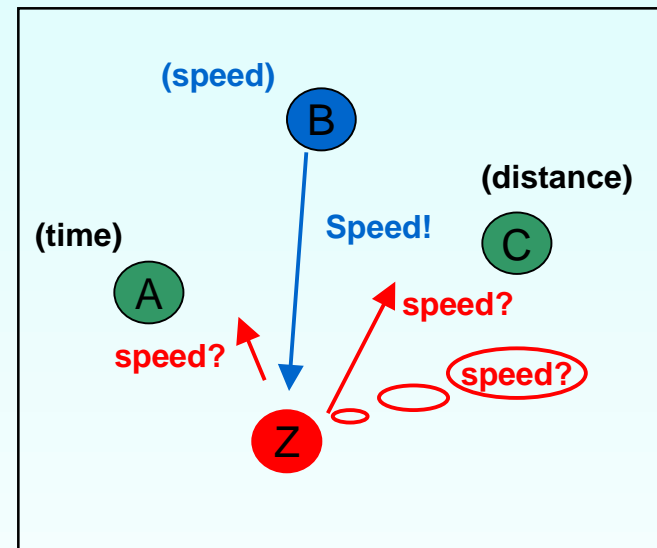
- ◆ A, B and C present
- Devices offer some kind of service
- Z joins the group
- Z requires **woof** service

## Service Discovery Models

- ◆ Which device is providing the required service?
- ◆ **Pull model:** query the environment when service required
  - **Distributed pull**
  - **Centralized pull**
- ◆ **Push model:** push service information into the environment
- ◆ **Hybrid models**

# Service Discovery, Distributed Pull

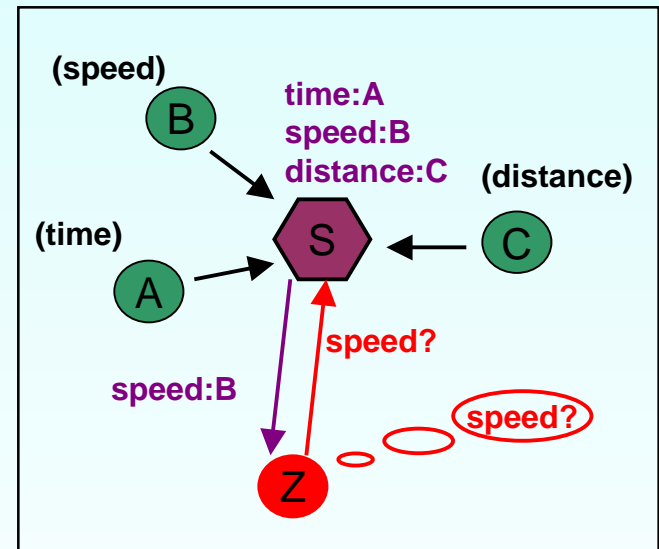
- ◆ Z requires **speed** service
- Z broadcasts request for **speed**
- B responds with service information
- Problems:
  - B in sleep mode?
  - Reachability
  - Service discovery slow
  - Broadcasts consume bandwidth



- Service Location Protocol (SLP)
- Universal Plug and Play (UPnP)
- Bluetooth's Service Discovery Protocol (SDP)

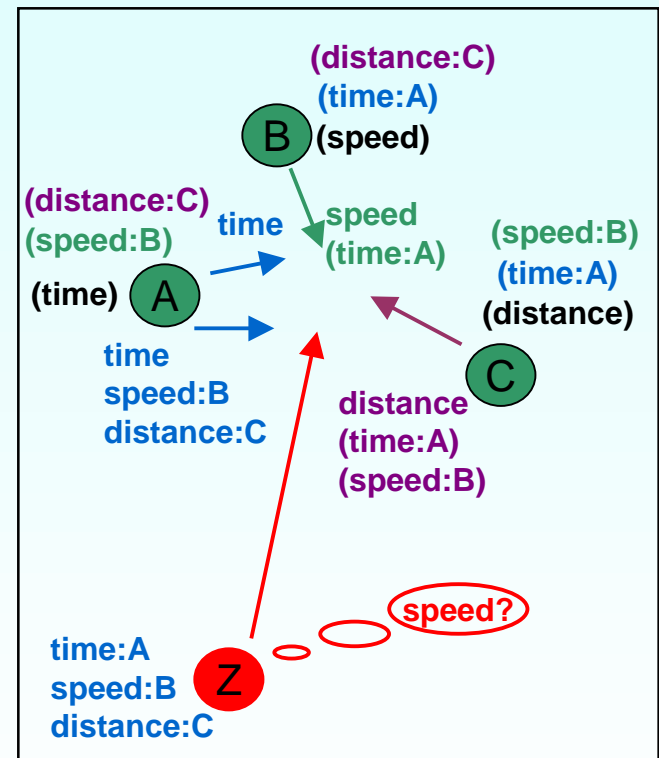
# Service Discovery, Centralized Pull

- ◆ Z requires **speed** service
- Central repository S
- A, B and C register with S
- Z queries S for **speed**
- S responds with service information
  
- Scalable
- Problem: Finding central repository
  
- Jini
- UPnP
- SLP



# Service Discovery, Push

- Nodes periodically broadcast service lists
- Received service lists update existing ones
- (New node) Z triggers rebroadcast
- Z updated
  
- Updated view of environment
- Problem: broadcast interval & flooding
- IBM DEAPspace



## Common Service Discovery Protocols - Revisited

- ◆ Sun's Jini, a java-based distributed model.
- ◆ IETF Service Location Protocol (SLP).
- ◆ Microsoft's UPnP.
- ◆ Bluetooth's Service Discovery Protocol (SDP).
  - These service discovery protocols cannot adapt to the changes and size of ad hoc networks quickly enough.

# Peer to Peer Models for Sensor Networks: Managing re-configurable networks

## Peer to peer Networking

- ◆ Another powerful networking technology that can support re-configurable nodes is a peer to peer network.
- ◆ In particular, it is one of the few approaches that could possibly support re-configurable sensor networks across the Internet.
- ◆ Peer to peer networks have a great deal of similarity to ad hoc networks but operate in an IP domain over an existing Internet infrastructure.
- ◆ Sensor networks require decentralized versions of spatial query for determining routes to sensors.

## The Scalability Problem

- ◆ **Why Gnutella Can't Scale. No, Really – Jordan Ritter (February 2001)**
  - <http://www.darkridge.com/~jpr5/doc/gnutella.html>
- ◆ Demonstrates mathematically that Gnutella bandwidth consumption goes up nearly 10 fold for each extra hop taken into account in the network.
  - For 5 qps  $N=8$  &  $T=8$  we get ~ **31.7GBps** of traffic.
- ◆ Fortunately GnutellaNet is not ideal and in fact natural partitions begin to form among participants such that the network exhibits properties of a “small-world” where cliques of connectivity are formed.
  - Scalability Issues in Large Peer-to-Peer Networks - A Case Study of Gnutella University of Cincinnati Technical Report 2001, M. A. Jovanovic et al.

## Clustering for Service Discovery

- ◆ Since then many papers have focused their attention on improving the search and routing algorithms for peer-to-peer computing.
- ◆ The JXTA platform, perhaps the most dominant development platform for peer to peer computing, supports a concept known as Peer groups and peer group advertisements for service discovery.

## JXTA Peer Groups

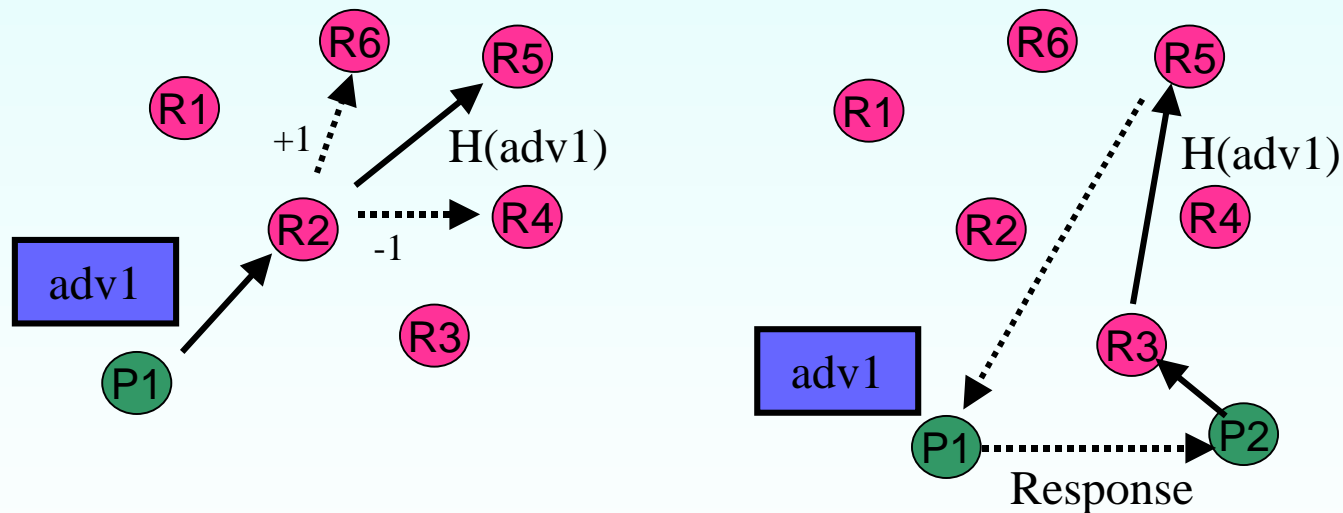
- ◆ Peer groups are identified via a unique peer group ID (GID).
- ◆ A module specification ID (MSID) points to an advertisement that describes all of the peer group services, a peer group name (Name) and description (Desc).
- ◆ All peer groups cache and publish and exchange advertisements to discover and find available network resources.
- ◆ Peers discover resources by searching for their associated advertisements.
- ◆ Advertisements are published with a TTL value.

## JXTA Super Peers – Dealing with scalability

- ◆ Super peers are default resolvers that have agreed to cache advertisement requests.
- ◆ Basically they are well known locations that are used for indexing and locating advertisements.
- ◆ JXTA 2.0 Supports a Concept known as rendezvous that provides a low level service discovery mechanism for large peer to peer networks.

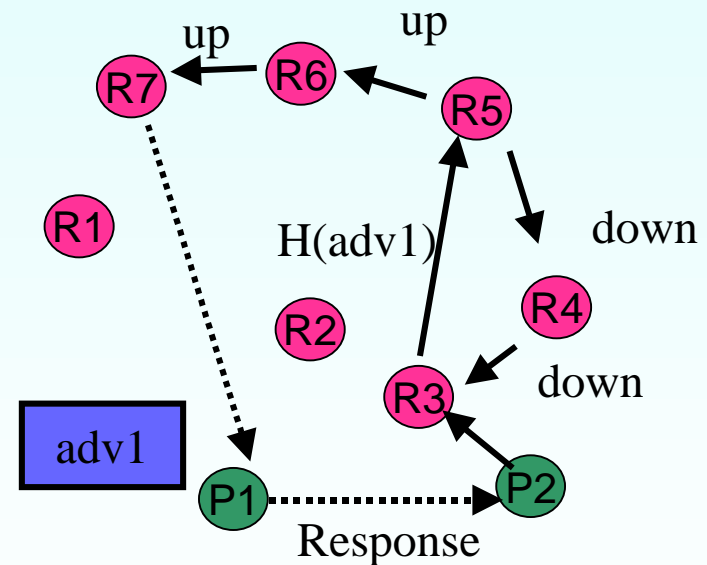
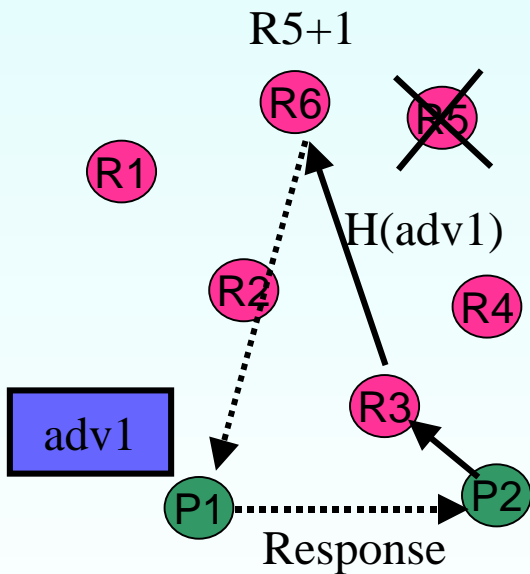
## JXTA Rendezvous Model

- ◆ Uses a loosely Distributed Hash Table (DHT) to index location of services.



# JXTA Rendezvous Model

◆ Suppose R5 goes down



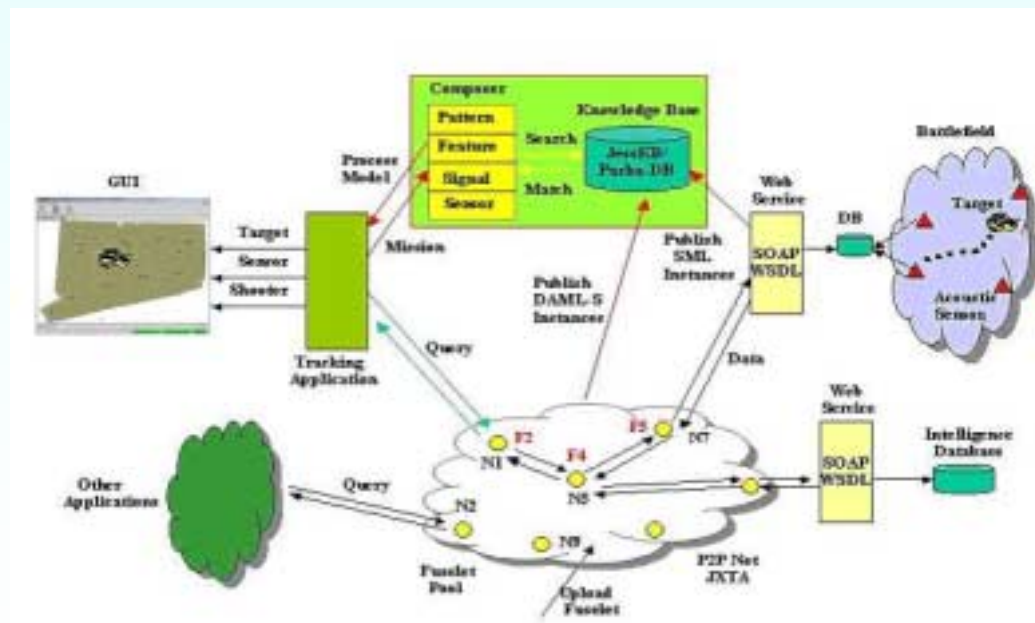
No Distributed  
Maintenance Required  
DHT +1 Offset

## JXTA Remarks

- ◆ A powerful framework developed by SUN Microsystems for the development of ad hoc peer to peer networks.
- ◆ Reference: Project JXTA 2.0 Super-Peer Virtual Network - Mike Duigou, Carl Haywood, Jean-Christophe Hugly, Eric Pouyoul, Bill Yeager, [www.jxta.org](http://www.jxta.org)

## Sensor Networks that use JXTA

- ◆ Dartmouth College - Tactical Sensor Networks (G. Jiang et al.) <http://actcomm.dartmouth.edu/~jiang/>
  - Combines high level semantic representations of services with re-configurable wireless sensor.



## Sensor Networks that use JXTA

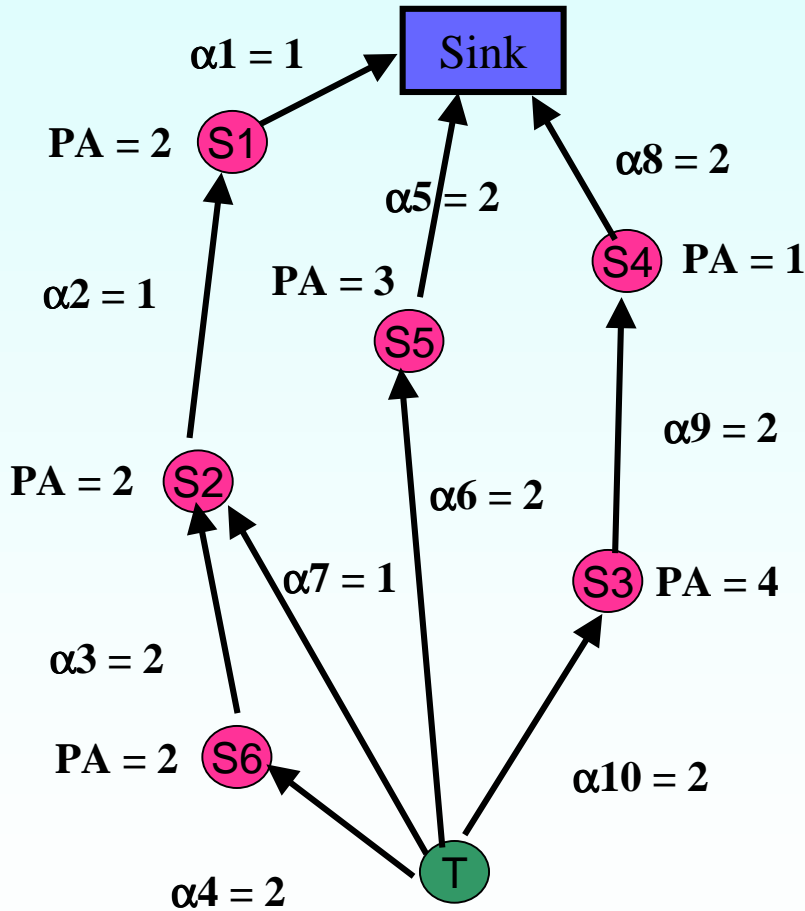
- ◆ Peer-to-Peer Spatial Queries in Sensor Networks (Murat Demirbas Hakan Ferhatosmanoglu) Ohio State University.
  - Developed a Peer Tree model for efficient distribution of service information in peer to peer sensor networks.

# Wireless Sensor Networks and Service Discovery

## S.D. in the Network Layer

- ◆ Most of this work is in the Network layer that investigated how to discover which nodes are relevant to communicate with.
- ◆ The key here is that Sensor Networks are data centric networks
  - Routing is based on more than addressing. More information is required in the query.
- ◆ Data aggregation is a key point
  - Clustering is important. Similar to peer to peer networks.

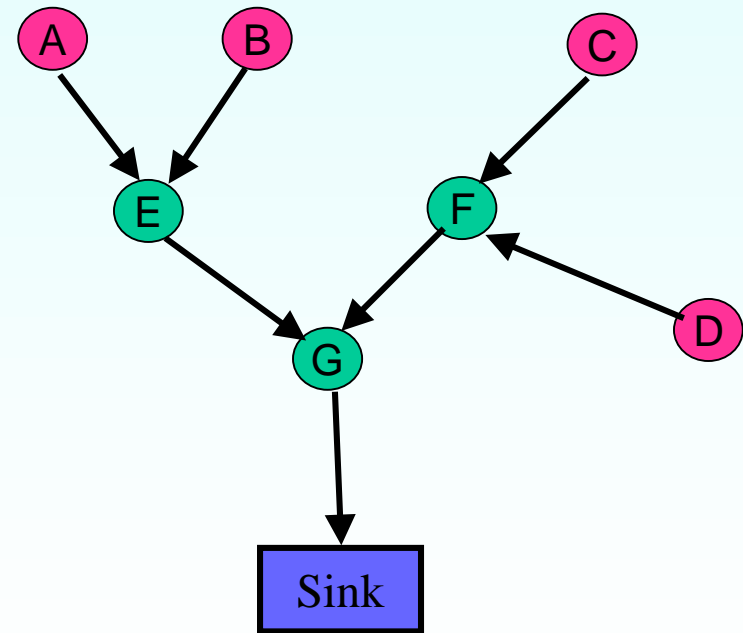
# Common Routing Algorithms



- **R1:** Sink-S1-S2-T, total PA = 4, total  $\alpha$  = 3
  - **R2:** Sink-S1-S2-S6-T, total PA = 6, total  $\alpha$  = 6
  - **R3:** Sink-S5-T, total PA = 3, total  $\alpha$  = 4
  - **R4:** Sink-S4-S3-T, total PA = 5, total  $\alpha$  = 6
- 
- **MAX PA:** Choose R4
  - **MIN ME:** Choose R1
  - **MIN HOP:** Choose R3

## Data Centric Routing

- ◆ Queries based on “area where temperature  $> 70$ . Routing algorithms appear more like trees to resolve implosion problem.



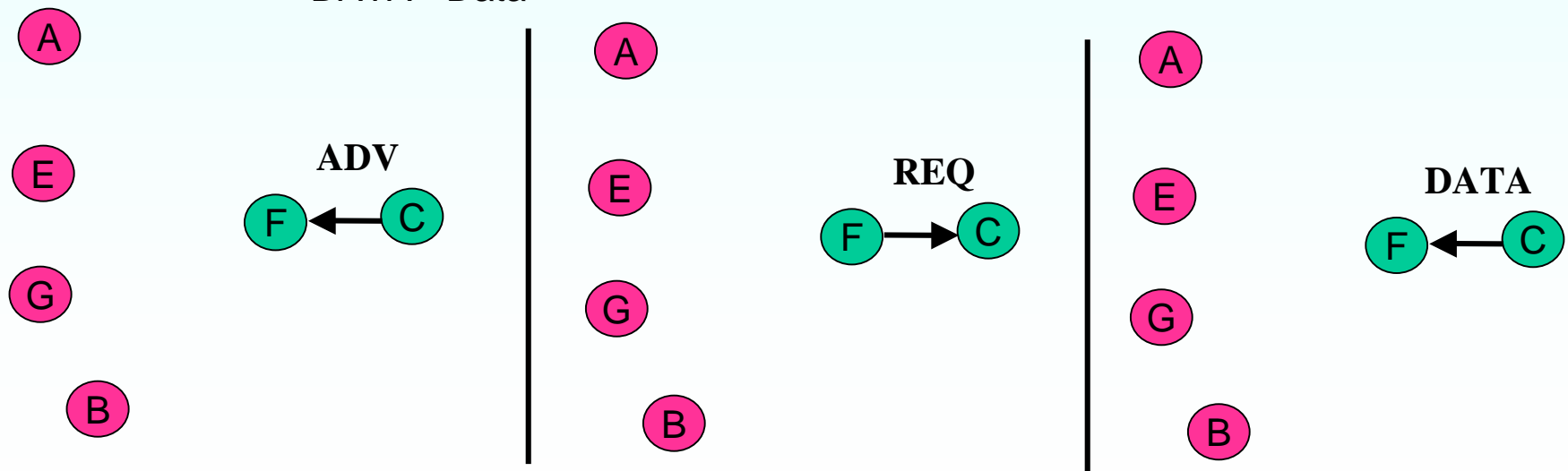
- *Data Fusion Techniques for Data Aggregation*

- *W. R. Heinzelman, J. Kulik, and H. Balakrishnan, “Adaptive Protocols for Information Dissemination in Wireless Sensor Networks,” Proc. ACM MobiCom '99, Seattle, WA, 1999, pp. 174–85.*

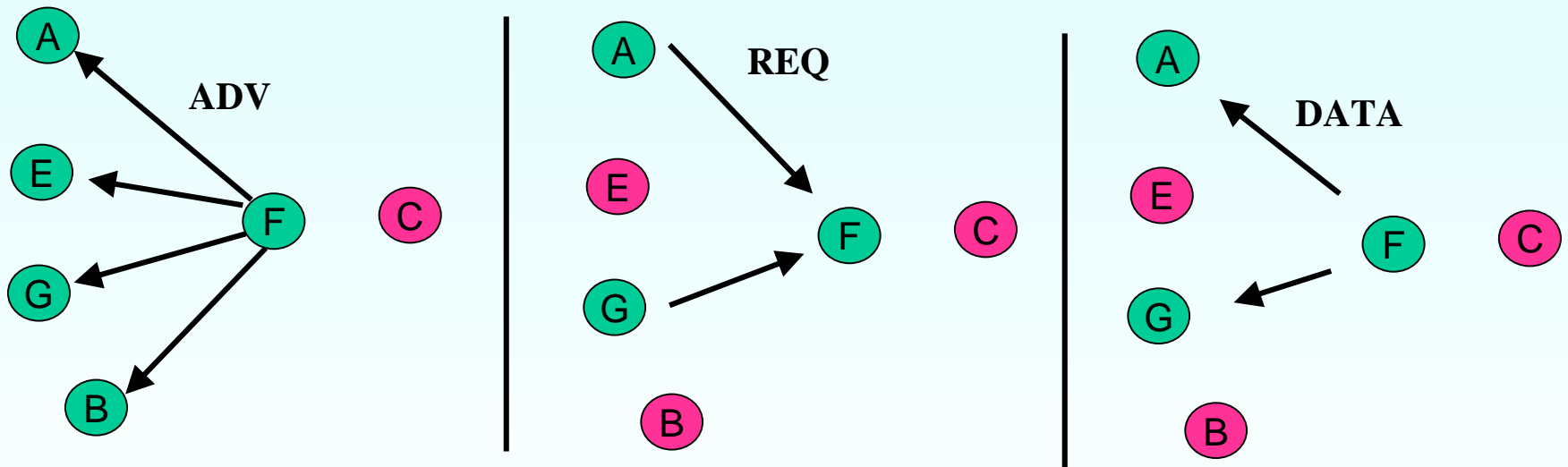
# Sensor Protocols for Information via Negotiation

◆ A family of protocols that reduce the bandwidth of sensory data by focusing on advertising information about the sensor rather than sending data (service discovery).

- 3 types of messages available:
  - ADV – Advertise
  - REQ – Request
  - DATA - Data



# Propagation of Sensor Service Information



## Directed Diffusion

- ◆ Sink sends out interest (Service Request with attributes)
- ◆ Each node can cache this request.
- ◆ Timestamp and gradient values are stored.
- ◆ Uses a local gradient approach to determine best path.
- ◆ When data is available the data is sent along the best gradient path.
  - C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks,” Proc. ACM MobiCom '00, Boston, MA, 2000, pp. 56–67.

## Recap

### ◆ Web-based sensor networks

- Offer rich models for sensor services based on XML, DAML and Semantic Web.
- Sensor service discovery based on Web client server protocols like WSDL, UDDI, and Client Server.

### ◆ Peer to peer sensor networks

- Support re-configurable networks that are caused by the coming and going of sensors and interested parties.
- Routing and service discovery are closely related.

### ◆ Wireless Ad hoc Networks

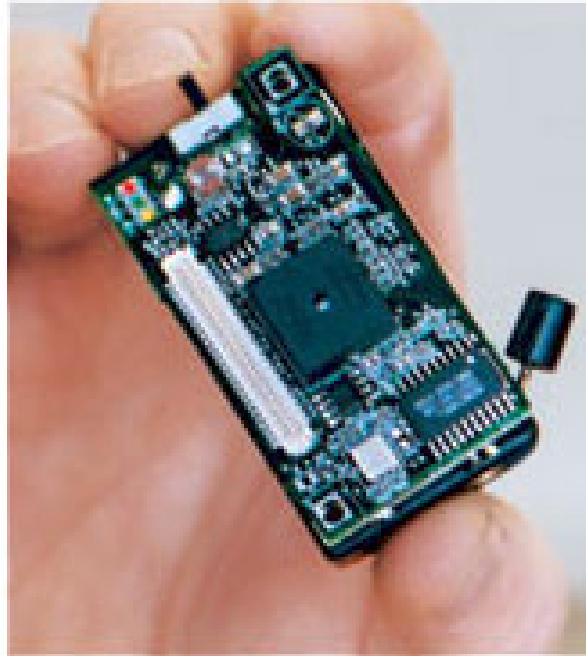
- Routing and sensor discovery are one and the same.

## Conclusion

- ◆ Sensor Networks cover a wide range of topics that range from Web-based sensor networks -> peer to peer sensor networks -> wireless sensor networks.
- ◆ Many wireless sensor publications never take into account anything to do with sensor models that might be crucial in an actual sensor network.
- ◆ Ad hoc networking (wired or wireless) integrate the relationship between service discovery and routing such that efficient service discovery and propagation techniques are key factors in the design of the sensor network.

## Example of Wireless Sensor Network Devices

- ◆ The UC Berkley Mote: Article from ICT Newsletter (January 2003 - Vol. 1 Issue 3)



**Mote maker: David Culler's "motes" monitor the environment and send reports wirelessly.**