Webdust

Spatial Web Overview

Rich Martin
Badri Nath
Rutgers University

DARPA site visit
8/1/2001

Spatial Web Goals

- **Describe objects and conditions in physical space**
  - What kind of tank last crossed the intersection?
  - How many are in the field?
  - Where's the projector?
- **Easy to add information**
  - Only require trust of either neighbors or a higher-authority
- **Allow wide range of data types**
  - want add info about different objects with different properties
- **Ad-Hoc construction**
  - System configures itself (within limits)
- **Not tied to any specific spatial model**
  - I don't think in WGS-84 coordinates (GPS)

Outline

- **Ideology and assumptions**
- **Example scenarios**
- **Realizing the software**
- **Challenges**
- **Timeline**

Spatial Web Ideology

- **Every physical object maintains a textual description of itself**
  - sensed data and object state - e.g. how much gas in the tank?
- **Objects are network addressable**
  - Contrast to diffusion routing with publish/subscribe addressability
  - How can a pub/sub model fit into natural notion of “physical object”?
- **Hierarchy of objects, servers and networks**
  - Move beyond the simple “flat sensor field” network & node assumption
    - Hierarchal tree-like structure more likely
  - Spatial servers and crawlers can leverage the hierarchy

The Spatial Web Concepts

- **SPatial ObjecT (SPOTs)**
  - A name-able entity in the physical space
  - reachable via a network
  - Contains: (1) data, (2) location, (3) links to nearby objects
- **SPatial tAG (SPAGs)**
  - The location information
  - A textual description of the space described in the SPOT
- **SPatial Links (SPLINKs)**
  - The link information
  - Describe relationship between SPOTS
    - Neighbors
    - Superspace
    - Subspace

Spatial Web Representation

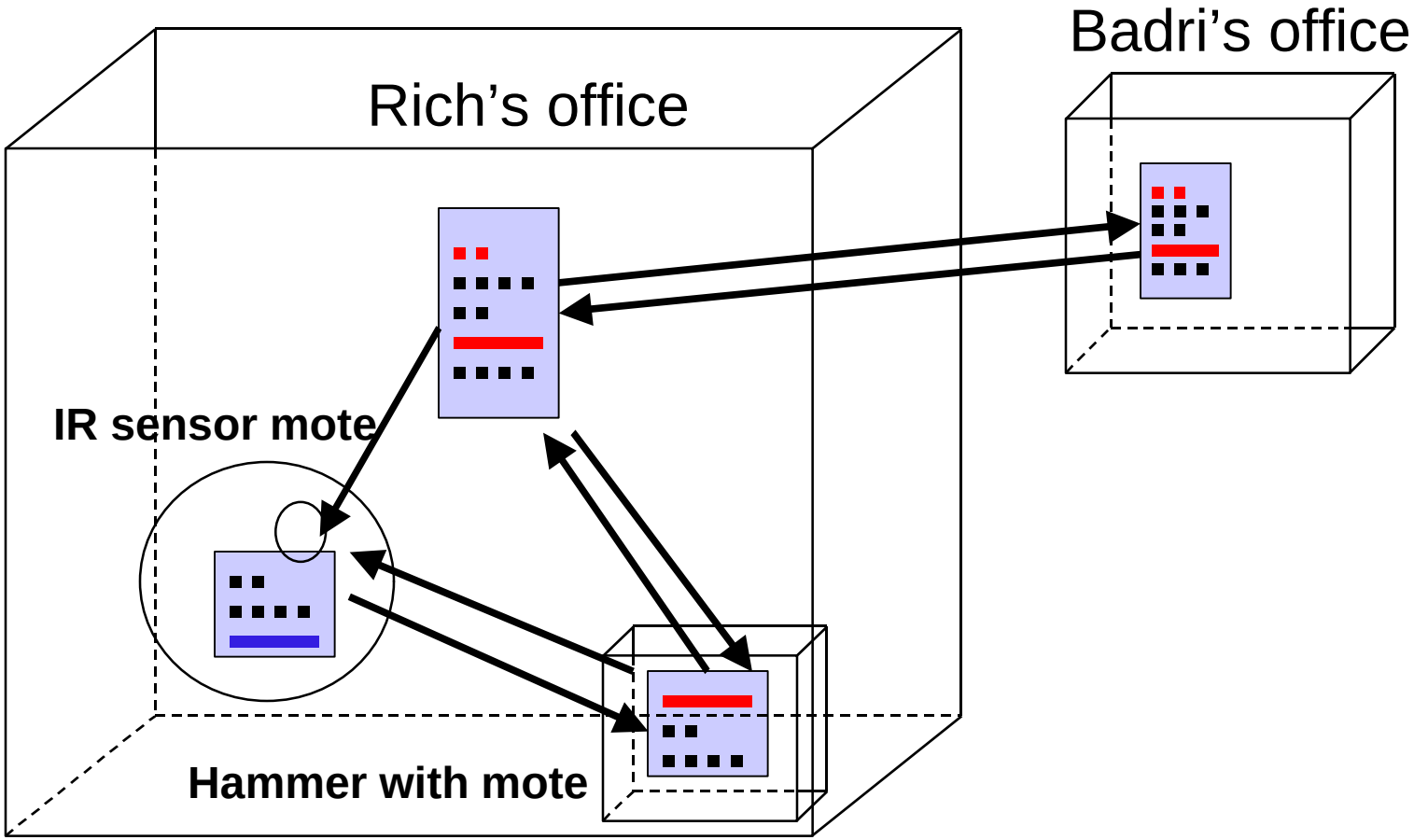
- **State of the physical space is defined as a distributed object graph**
  - Web, DNS maintain large distributed graphs
- **Key to success is how the graph can be extracted and analyzed**
  - Leverage wealth of graph theory on structure & traversal
  - E.g. Speed of extraction => nodes visited per time,
  - E.g. Energy consumed => watts/node
- **Mobility alters the link structure**
  - How fast can we detect these changes?
  - Staleness of info?
  - Pro-active updates?

Roots from 2 large distributed Systems

- **DNS (Domain Name Service)**
  - \+ Distributed authority & control
  - \+ Hierarchical naming & lookup
  - – Hard to add/remove content (DNS records)

- **WWW (World Wide Web)**
  - \+ Multiple data types
  - \+ Ad-Hoc structure
  - \+ Easy to add/remove documents
  - – No Hierarchy
  - – Very weak authority

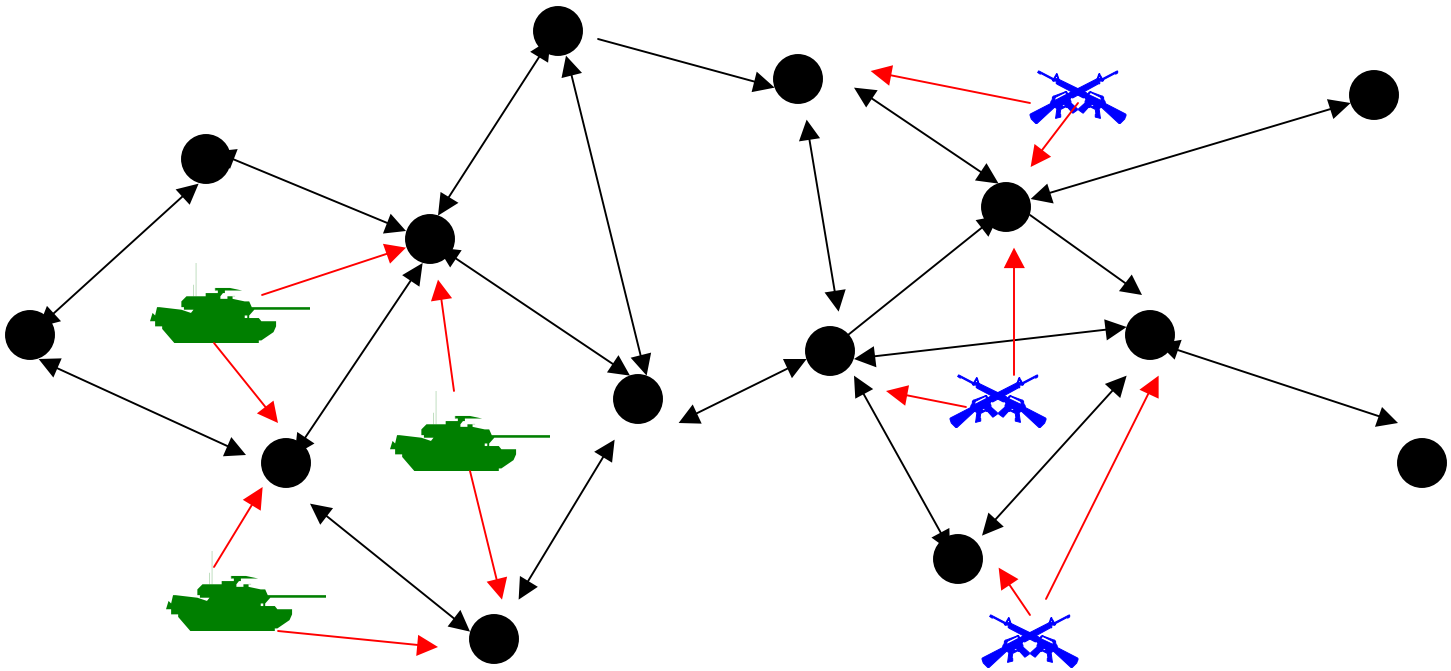
Example Scenario



Mobile Example

Stationary sensor objects define a stable graph

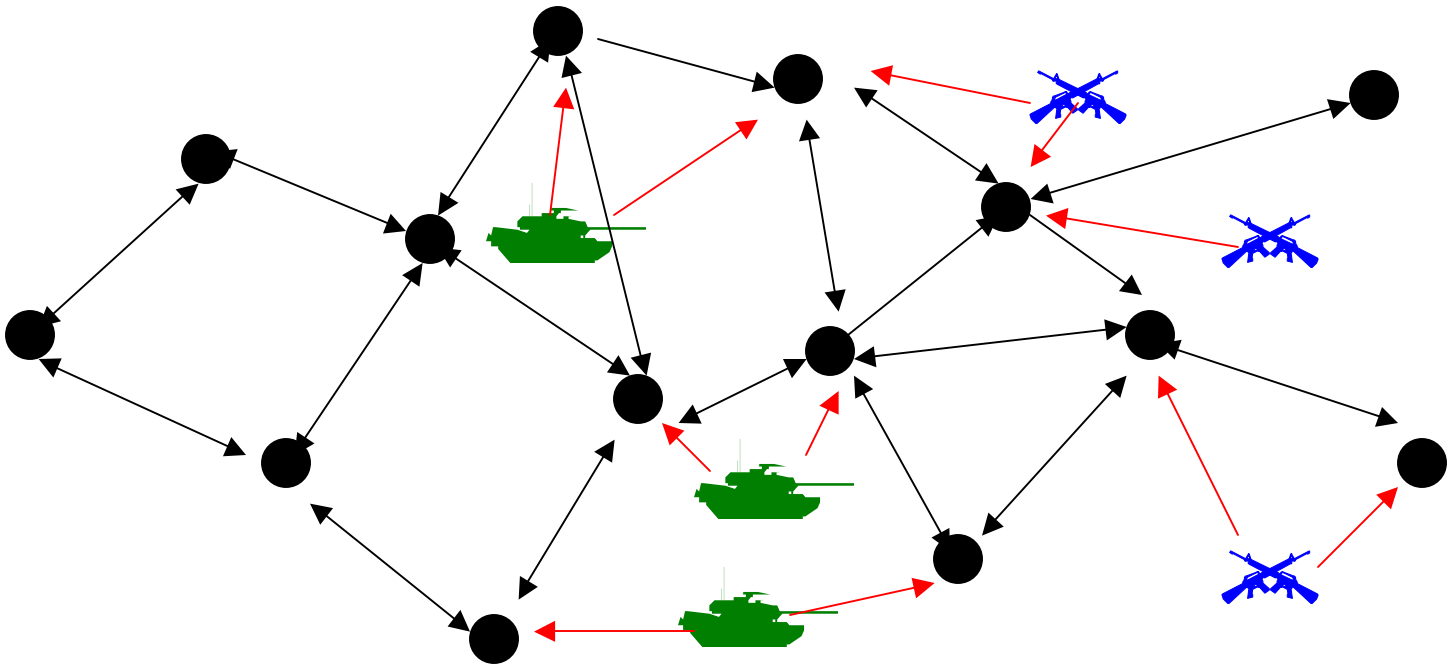
Mobile objects change link structure



Mobile Example

Stationary sensor objects define a stable graph

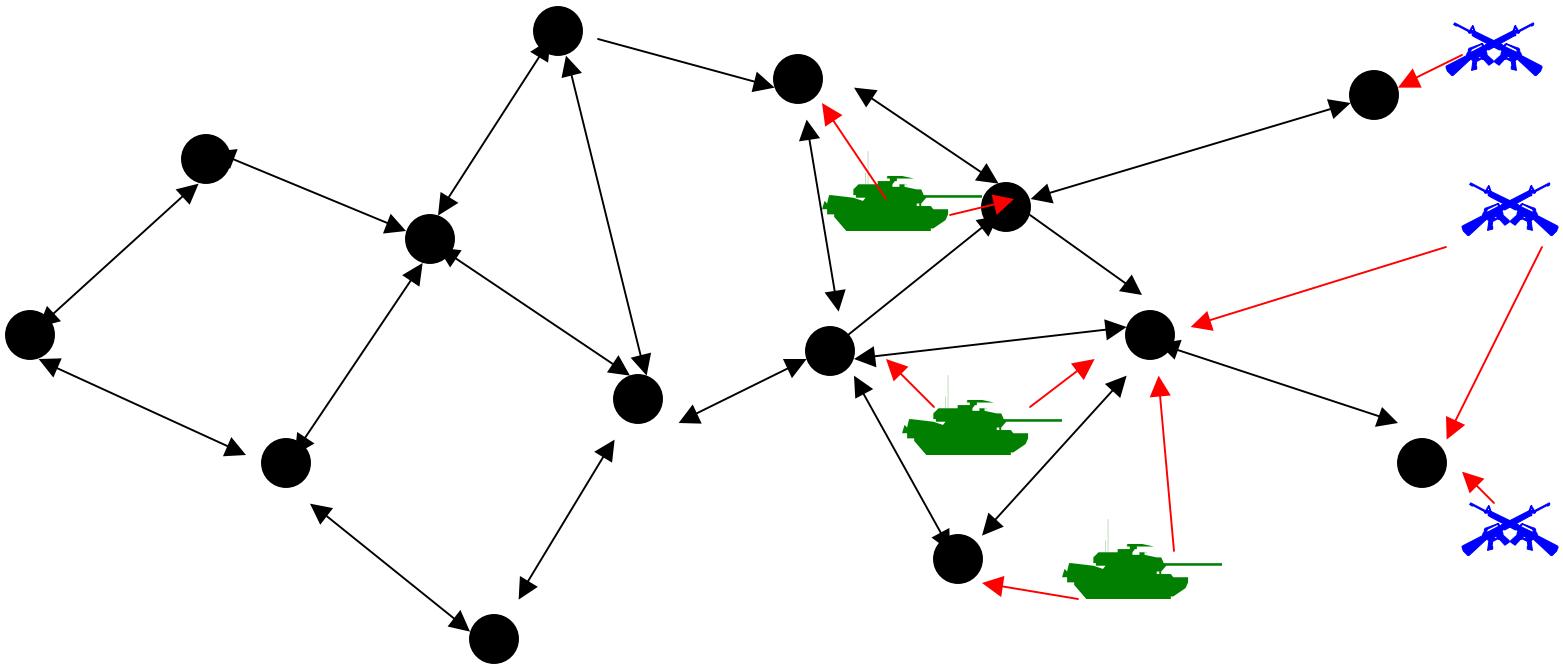
Mobile objects change link structure



Mobile Example

Stationary sensor objects define a stable graph

Mobile objects change link structure



Realizing the Spatial Web

- **3 Software components:**
  - Spatial Object
  - Crawler
  - Server
- **Use "HTTP-lite" protocol to "glue" components together**
- **2 Protocols**
  - peer discovery protocol
  - SPOT transfer protocol

Example SPOT

```
<SPOT>
   <name>Rich's Martin's office</name>
   <frame>WGS-84</frame>
   <spag> <cube lat="+40.52130",
                lon="-74.46103",
                alt="62m",
                side="3m"> </spag>
   <splink> <sub
   ="http://128.6.4.4:/sp1"> </splink>
</SPOT>
```

SPOT data

frame of reference

shape & position

spatial link

Example Linkage

superspace link

```
<SPOT>
 <name>Rich's Martin's office</name>
<frame>WGS-84</frame>
<spag> <cube lat="+40.52130",
                lon="-74.46103",
                alt="62m",
                side="3m"> </spag>
                <splink> <sub "http://128.6.4.4:/sp1">
                </splink>
</SPOT>
```

```
<SPOT>
    <name>Rich's claw hammer</name>
    <frame>Core Hall </frame>
    <spag>Room 304</spag>
    <splink><sup="http://www.cs.rutgers.
    edu/~rmartin/office"> <splink>
</SPOT>
```

subspace link

SPAGs

- **A short textual description of the shape & location of the objects**
  - Both shape and position needed!
- **Coordinates dependent upon the SPOT's frame (E.g. datum)**
  - E.g. WGS-84 for GPS, Military Grid Reference System (MGRS), even own frames (E.g. "core hall")
- **Few assuptions make it easy to create new SPOTs**
  - most people don't think in MGRS or WGS-84
- **Spatial Server does hard work of combining SPAGs**
  - Assume server is running on a "powerful" node
    - full blown OS, database, floating point, stable storage …
  - Today, even an Ipaq is good enough!

SPLINKs

- **A URL-style pointer to another SPOT**
  - Superspace , pointer to enclosing SPOT
  - Subspace, pointer to enclosed SPOT
  - Neighbor, pointer to non enclosing/overlapping SPOT
- **DNS-like defined hierarchy aids crawling SPOTs**
  - ignore regions, sampling regions, directed crawls – hard on the web!
- **Spatial graph structure leverages network topology**
  - network structure at low/med grain enforces spatial structure
  - peer discovery protocol enforces at the lowest level
  - Higher levels enforced by natural structure created by humans

Spatial Crawlers and Servers

- **Spatial Web Crawler**
  - Charged with bringing SPOTs to the server(s)
    - I.e. load the graph into the server
  - More structure than a web page for intelligent traversal
  - Defining speed and range of crawling are key research

- **Spatial Web Servers**
  - Hold a datastore of SPOTs
  - Fit SPOTs from different frames into coherent whole
  - Answers spatial queries
  - Can mass-exchange records (SPOT-transfer) with other servers

SPOT peer discovery protocol

- **SPOT peer discovery protocol used to create ad-hoc spatial structure**
  1. Local Broadcast
  2. Neighbors respond with SPOTS
  3. Add Splinks to the local SPOT

- **Add-hoc link structure should match natural spatial structure defined by wireless and wired links**
  - Aggregation of local area networks map cleanly to spatial structure

SPOT transfer protocol

- **Method to aggregate SPOTs and manage size and scope**
- **A crawler collects SPOTs over a given region**
  - entire world not reachable
- **Server pieces together into a searchable database**
- **SPOT-transfer protocol moves SPOTS between servers**
  - E.g. Give me all SPOTs within cube X,Y,Z, S
  - Like DNS “zone-transfer”, no web equivalent
- **Leverages natural hierarchy**
  - Aids security (maybe hinder if not careful!)
  - Aids manageability

Security

- **Security: How do I know the data isn't tamered?**
  - SPOT encryption
  - Challenge protocol for servers
- **Authority: How is the data coming from the "owner" of a space?**
  - "web of trust" model for links, prevent bogus regions of spatial web
  - Signed SPOTS

Just a few of the challenges …

- **Tradeoff between accuracy, staleness and power consumption**
  - How many objects can the crawlers visit? Statisticly sample?
  - When to locally crawl for data or look up in a spatial server?
  - Mobility disrupts the graph.

- **Addressability of sensors a assumption valid?**
  - Should a single sensor map to a single SPOTs?
  - Will hierarchical data save us?

- **Security and Authentication**
  - Handle broken/bogus crawlers
  - Can SPOTs be made secure?
  - Can we protect against bogus SPOTs?
  - What if a region is compromised? (Like search-engine spam)

Timeline

- **Past Year:**
  - Version 0 of the Spatial web, SPOT, SPAGs defined
  - initial crawler
  - No spatial DB
- **This Year (2001-2002)**
  - Cram a SPOT onto motes
  - directed crawler and spatial DB
  - Use sensoria nodes as level 2, execute crawler and spatial DB
- **Next Year (2002-2003)**
  - SPOT transfer protocol
  - Experiments to measure crawl rate, time and power constrained crawling

Backup sides

Why not “just use a DMBS”

- **Static Schema**
  - Hard to add new data-types
- **Force used of SQL-query languages**
  - What if my app goes from SPOTs -> visualization directly
- **Locally rapidly changing state**
  - Spatial web approach keeps them localized
  - Why update the DB? Just leave on the object and uncrawled
- **Spatial Server probably will use a DMBS**
  - Centrally administered keeps the data