Coexistence, Collaboration, and Coordination Paradigms in the Presence of Mobility

Gruia-Catalin Roman

Department of Computer Science and Engineering
Mobile Computing Laboratory

4 April 2008

Washington University in St. Louis

Presentation Outline

- Presentation theme

- Prevailing trends

- Shifting foundations

  - Protocols

  - Algorithms

  - Formal models

- Paradigm shifts in middleware design

  - Coordination

  - Coexistence

  - Collaboration

- Conclusions

Presentation Theme

- *Computing and communication in the presence of mobility demands a new way of thinking*
  - Changes often invalidate fundamental assumptions
    - End of the cold war (1989)
  - We are on the cusp of a major technological and social realignment
  - Predicting the future is a risky proposition
    - Telephone (1876)
    - Internet (1969)

Prevailing Trends

… in which we learn that we are all in this together

Technology in Support of Mobility

- Miniaturization
- Wireless communication
- Localization technology
- Sensing devices
- Battery technology

- Code mobility

- Nomadic networks
- Ad-hoc networks
- Sensor networks

The Social Dimension

- A society on the go
- Growing expectations
  - Increasing reliance on information technology
  - Integration of computing and communication into the fabric of society
- Market acceptability and adoption emerging as powerful forces in technology development
  - Government policies and regulations unable to keep pace with technological advances

Application Development Opportunities

- Fire tracking and monitoring
- Ambulatory patient monitoring
- Container tracking
- Car to car interactions on highways
- Sensor assisted robot navigation

- Contents delivery on the phone
- Self managing assembly line
- Assisted airplane inspection and repairs
- Museum visit and city tours
- Disaster response
- Nature exploration
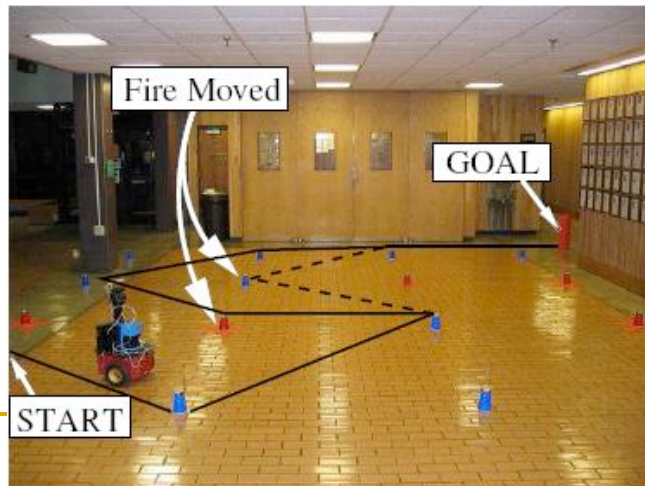
Shifting Foundations

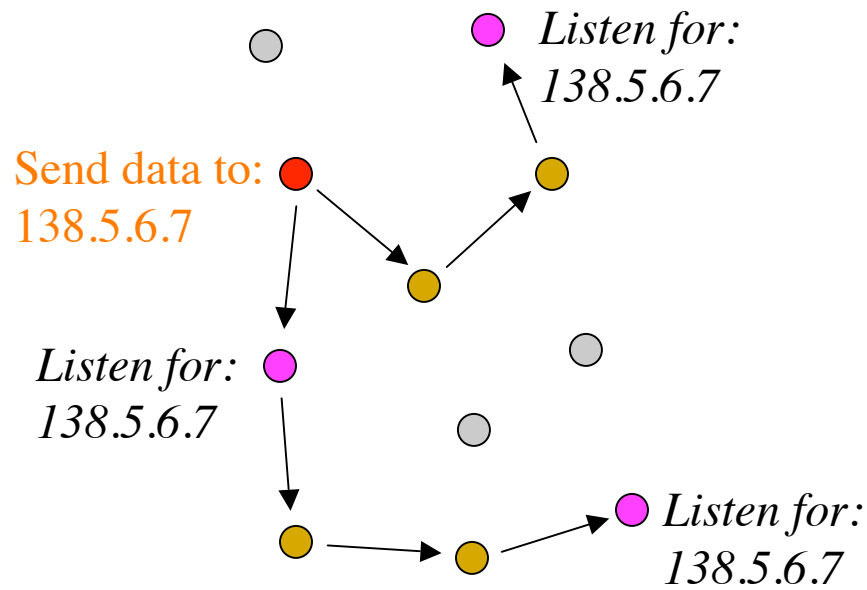*… where we find out that there are things our teachers never told us*

Protocol Design

… about sending messages to our friends and how to keep the highway clear
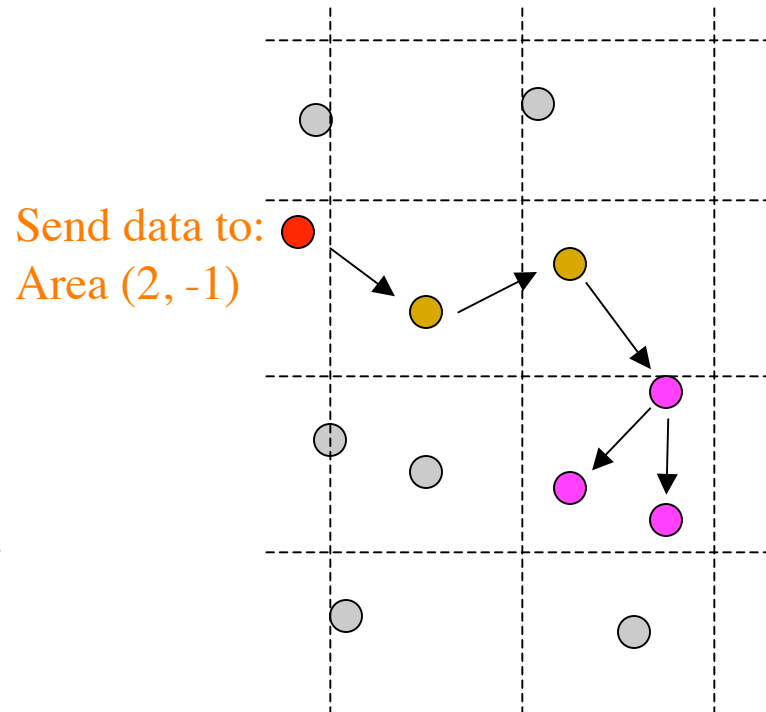
Multicast Revisited

- A multicast group is a set of nodes known to the world by a shared name

- A spanning tree that includes all group members is constructed and maintained

- Nodes may join and leave the group

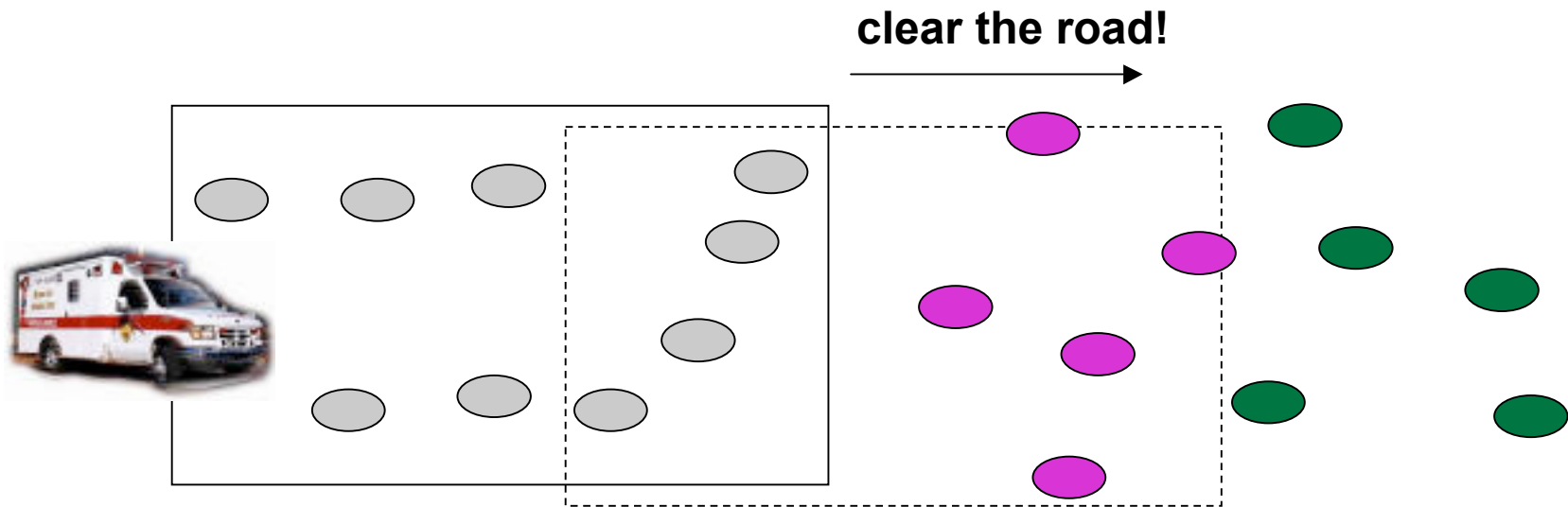- Data is delivered to all group members

Multicast Paradigms

Listen for:
138.5.6.7

Send data to:
138.5.6.7

Listen for:
138.5.6.7

Listen for:
138.5.6.7

Send data to:
Area (2, -1)
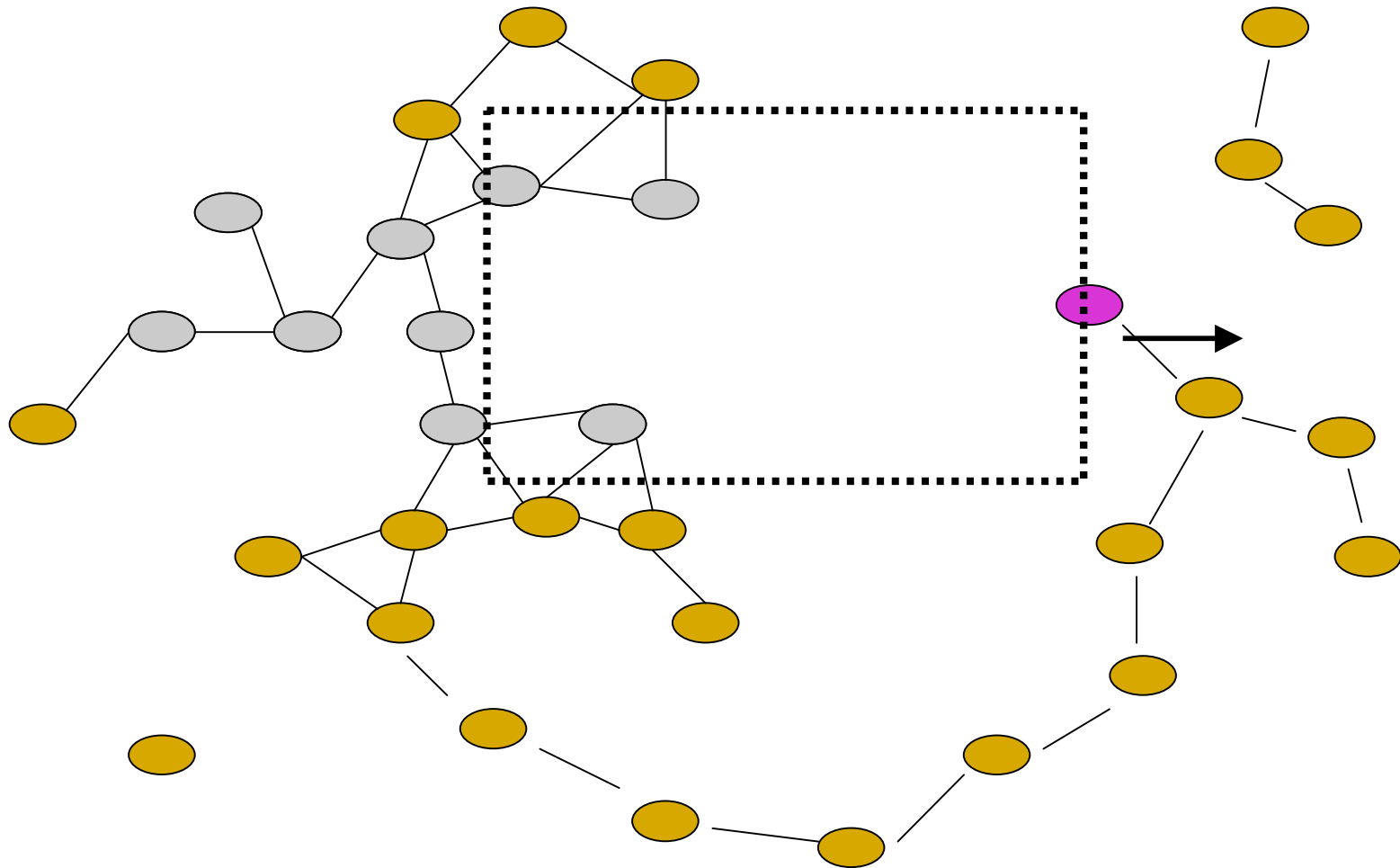
Standard Multicast

Geocast

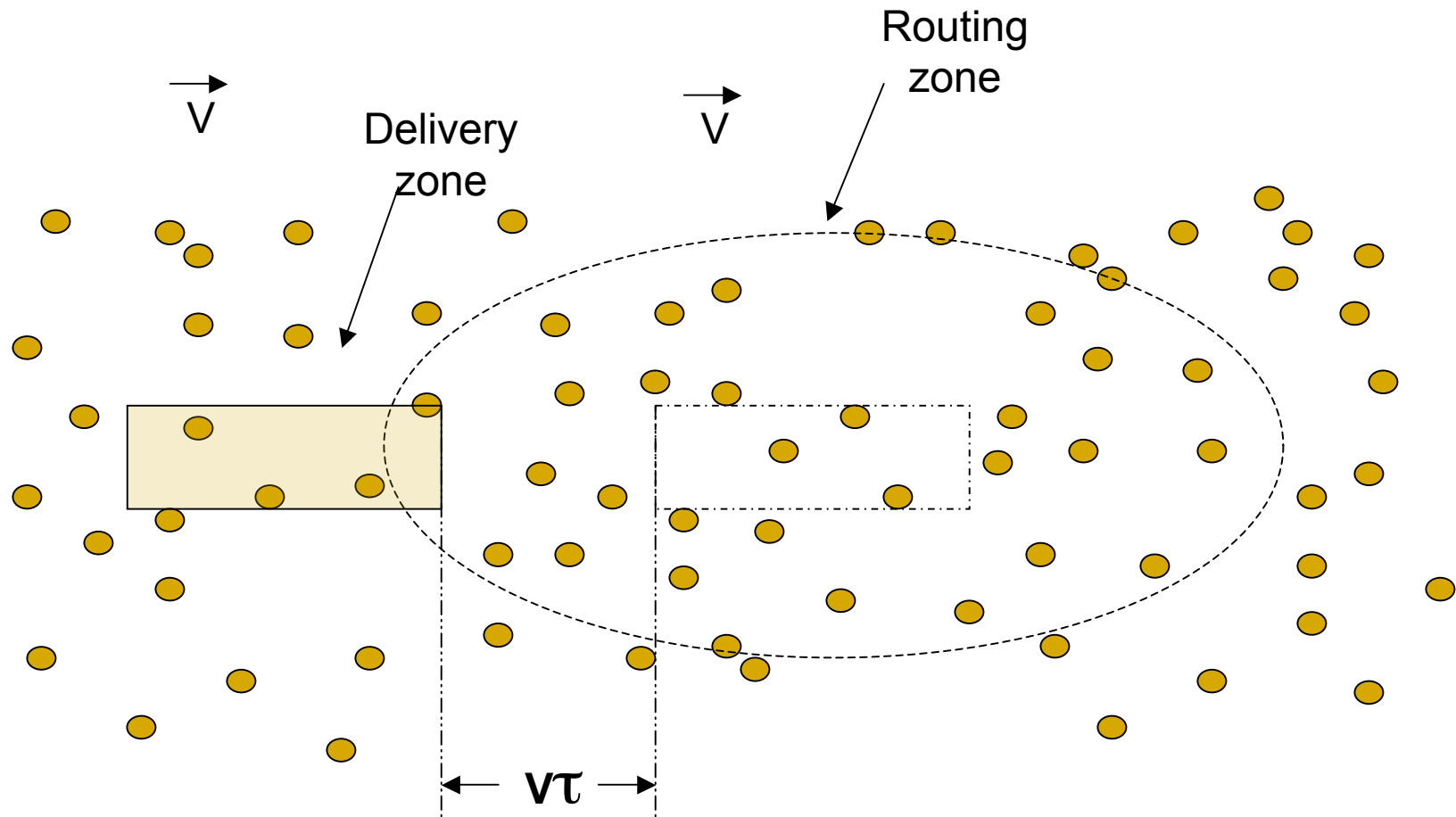# Mobicast: Spatiotemporal Multicast

- Just in time message delivery along a specified trajectory
  - Ambulance warning
  - Intrusion detection

clear the road!

Technical Challenges

A Mobicast Protocol Overview

Routing zone

$\vec{V}$

$\vec{V}$

Delivery zone

$v\tau$

Lessons Learned

- **Mobility changes the questions we ask**
  - mobile query—data prefetching and just in time delivery
  - location query and tracking
- **Spatiotemporal constraints alter our perspective**
- **Geometric characterization is a useful tool**
  - measures of sparseness
  - face aware routing
- **Energy conservation impacts the solution space**
  - sleep scheduling
  - message release order
 - workload shaping

Algorithms

… where we discover that keeping track of things is a messy business

Termination Detection

- **Diffusing computations are a special case**
  - one source of activity
  - active nodes can wake up other nodes
  - nodes may go idle at any time
- **Sample solutions**
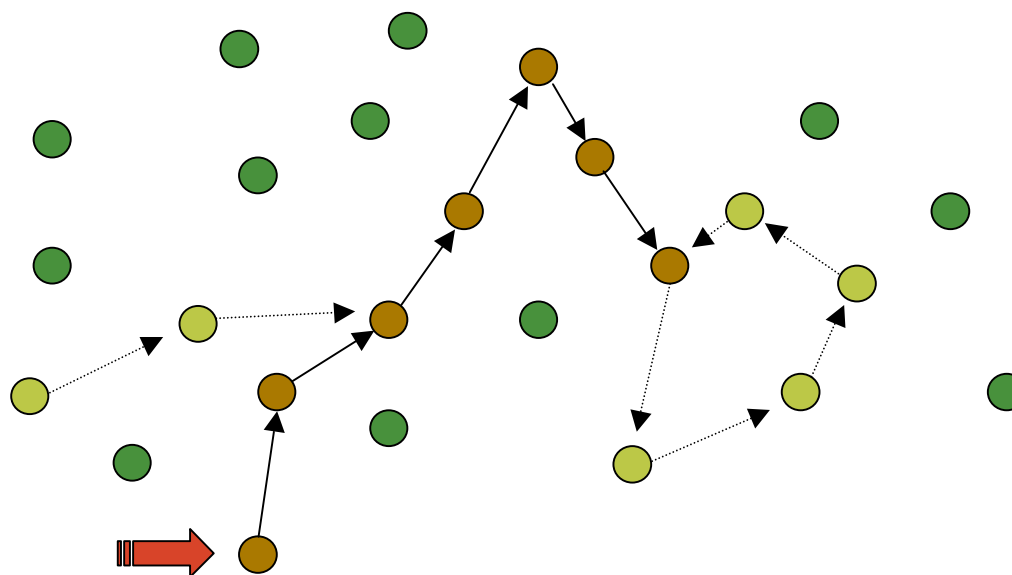 - counting
  - weight throwing
  - activation tree

Ad Hoc Network

- Migration of termination records
- Opportunistic routing
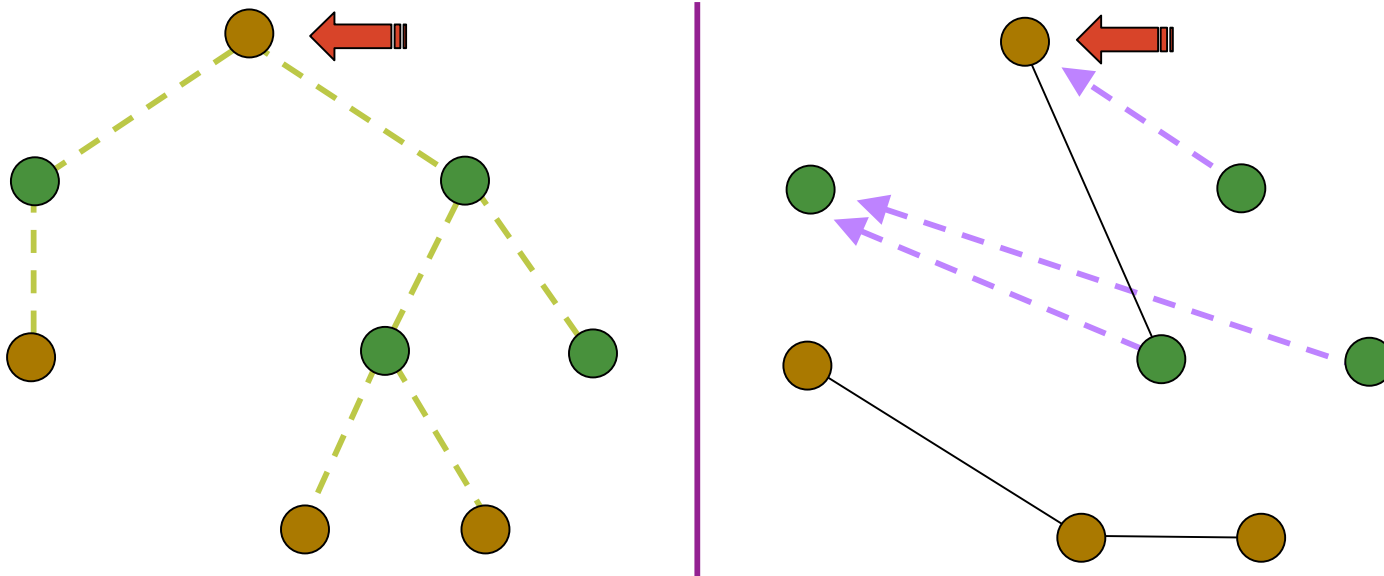
Wireless Cellular Network

- Tracking strategy
- Mobile-as-message model

Lessons Learned

- **New concepts may be needed**
  - weak vs strong termination
- **Guarantees may need to be conditional**
- **Models may facilitate translation of knowledge**
  - distributed algorithms recycled
- **Communication may take place over disconnected routes**
  - forwarding based on partial order
  - exploitation of motion profiles

Formal Models

… where the variable x gets the value v and sets an example for future things to come or go

UNITY

- ## The essence of concurrency

  - assignment statement

  - program as set of statements

  - nondeterministic selection

  - composition as set union

- ## Producer/Consumer

  P ::   x := x+1   if y = x

  Q ::   y := y+1   if y ≠ x

Mobile UNITY

- ## The essence of mobility in open environments
  - location as a distinguished variable
  - motion reduced to value assignment
  - composition as set union plus interactions
- ## Producer/Consumer

  P ::   x := x+1

  Q ::   y := y+1   if b

  Q.b := (P.x > Q.y) ^ (P.λ = Q.λ) reacts to true

Lessons Learned

- **Modeling open systems**
  - power of the quantifier
  - coordination dimension
- **Importance of conditional proofs**
  - Mobile IP
- **Expressive power shaped by real problems**
  - complex high level interactions
  - mobile code
  - fine grained mobility
 - Context UNITY

Paradigm Shifts in Middleware Design

… in which we learn that one must have software to make software, a bit like making money

Coordination

… where a girl never finds out how to make the pie, and we discover that splitting a pie is in the eye of the beholder
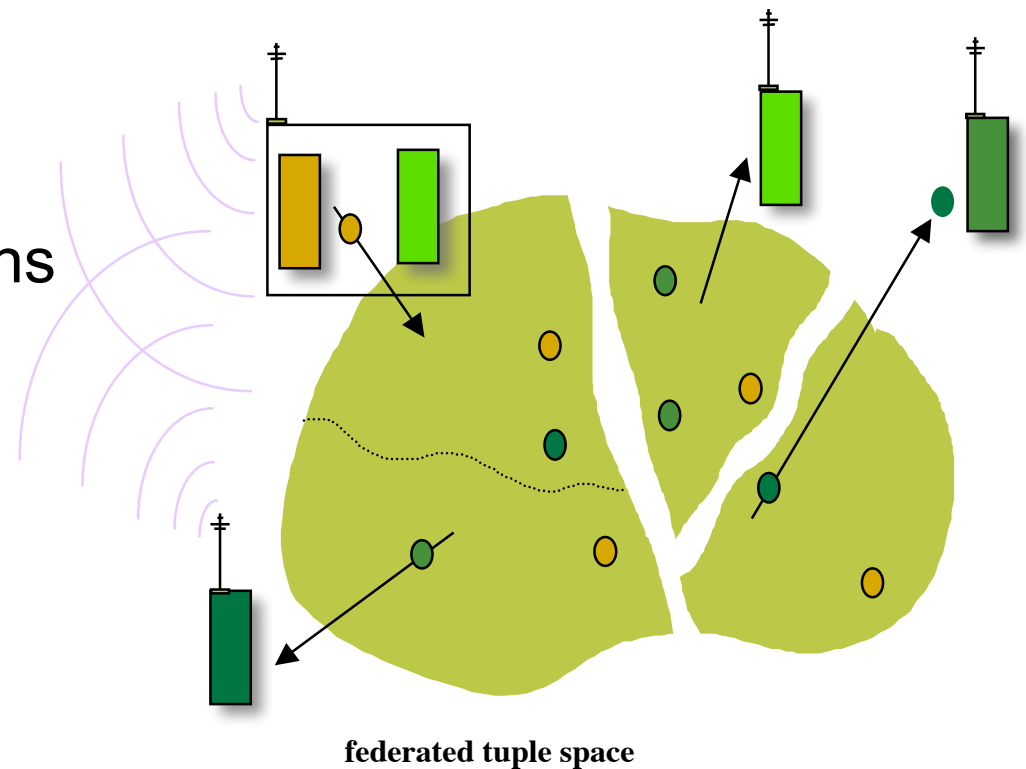
Linda

- A global persistent tuple space

- Three primitive operations:
  out(tuple); in(pattern); rd(pattern)
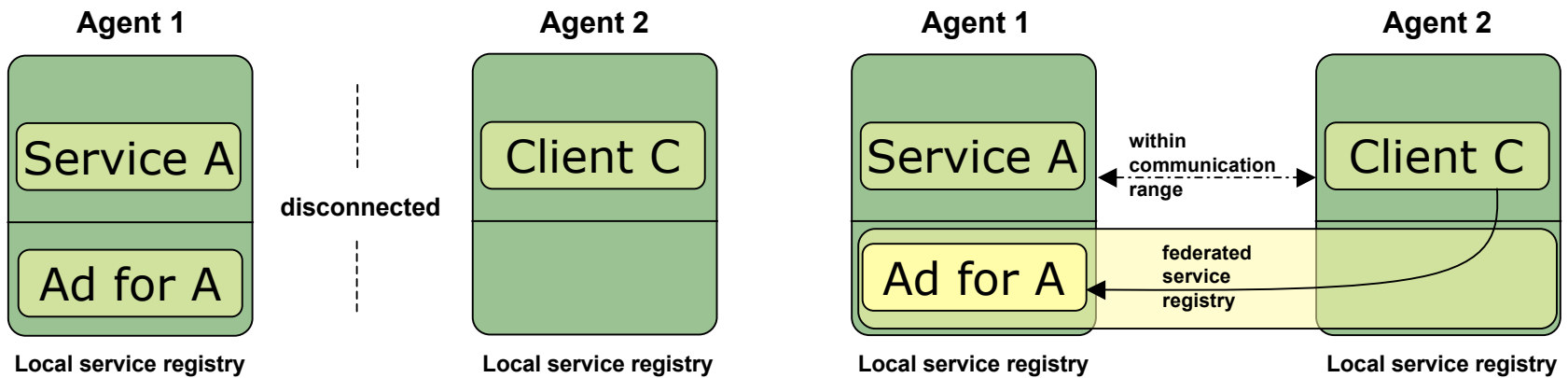
- Decoupled computing

- Concurrency at minimal cost
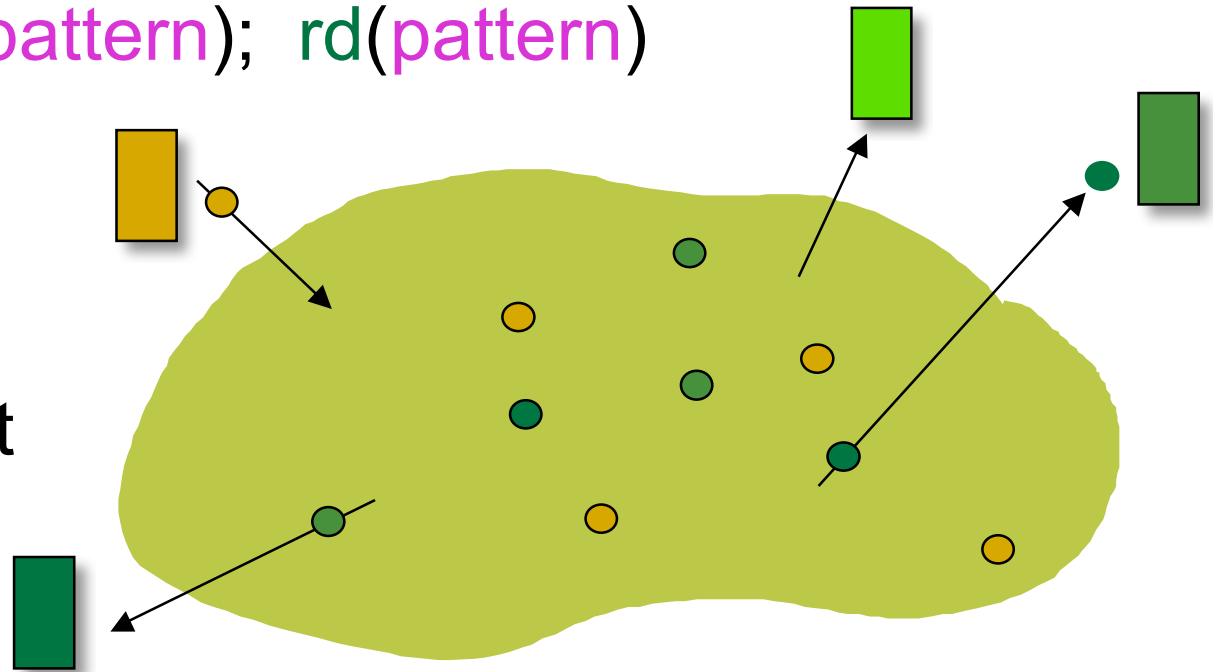
Lime

- **Agents can move among hosts and own tuple spaces**

- **Support for basic operations plus reactions**

- **Communication defines connectivity**

- **Hosts within communication range share data**

federated tuple space

Service Discovery

Agent 1   **Agent 2**

Service A   *disconnected*   Client C

Ad for A

Local service registry   Local service registry

Agent 1   **Agent 2**

Service A   **within communication range**   Client C

Ad for A   **federated service registry**
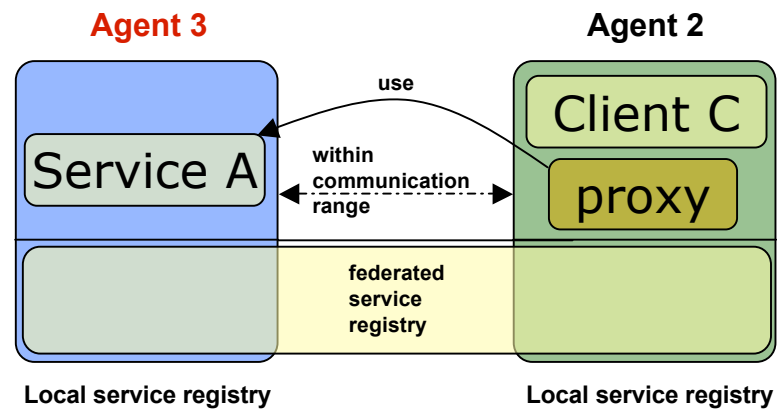
Local service registry   Local service registry

- The two hosts are too far away to communicate

- The two hosts establish contact
- Agent 2 can discover service A on Agent 1
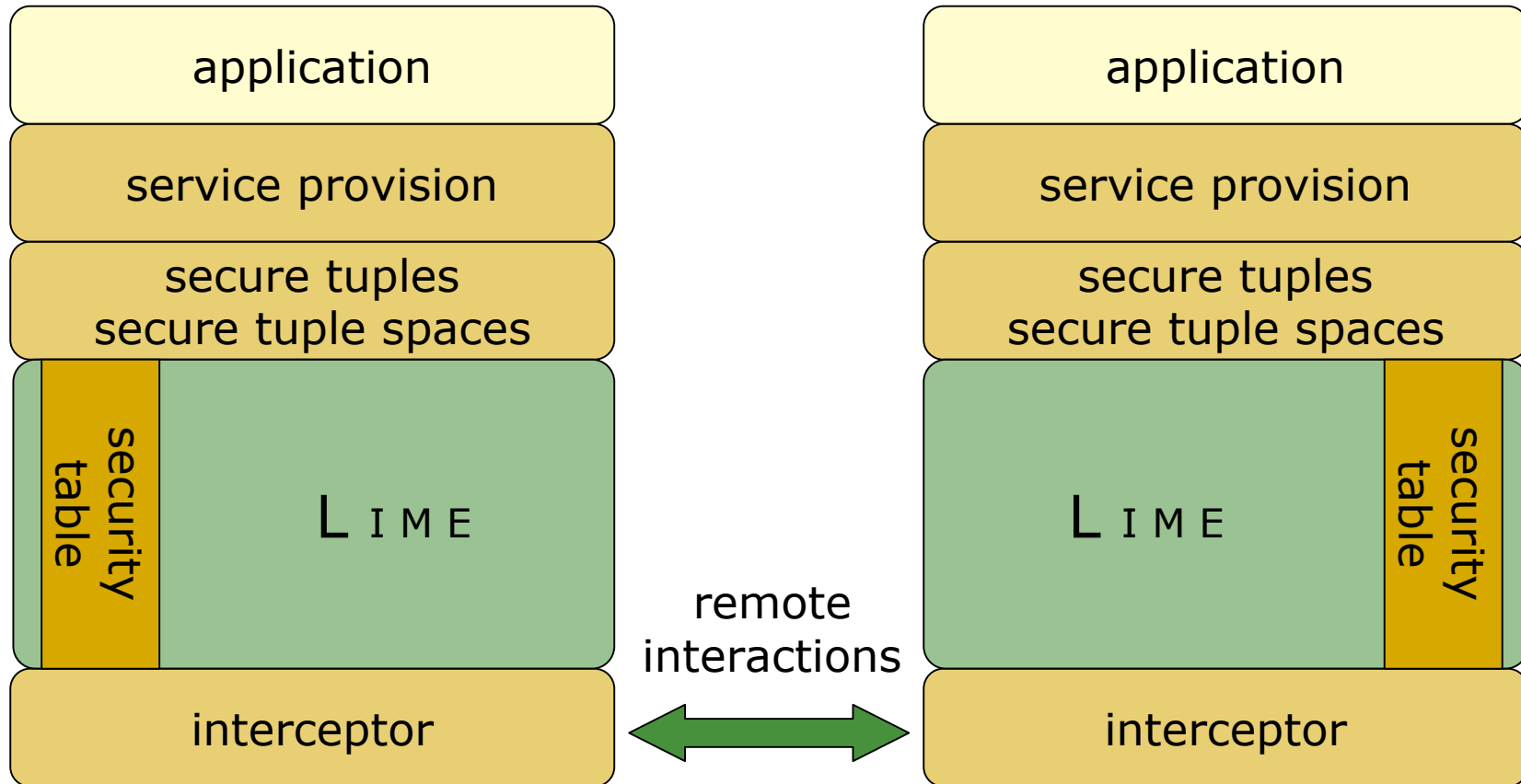
Service Utilization
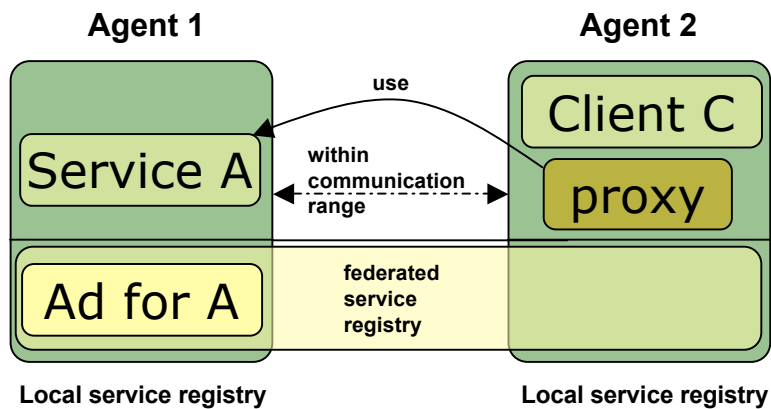
- Agent 2 uses service A offered by Agent 1

- Agent 2 continues to use service A which migrated to Agent 3
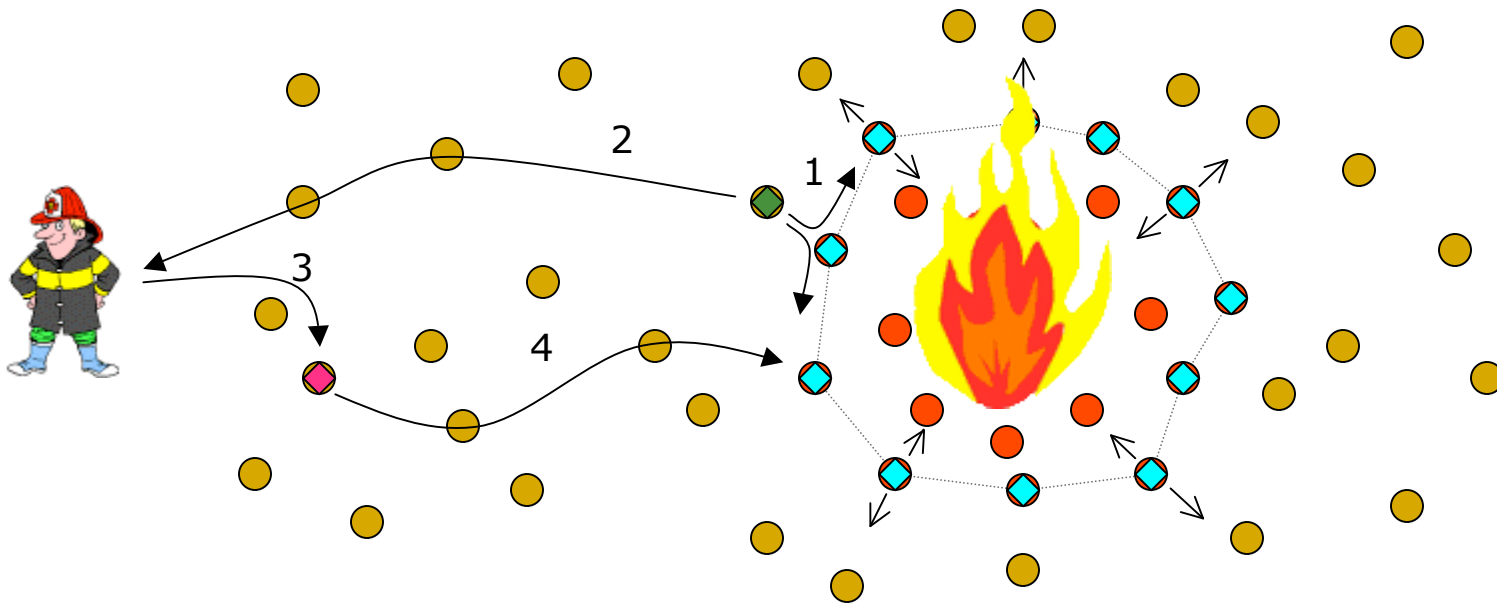
Security Enhancements

Lessons Learned

- **Lime**
  - specialization to mobility reduces development time
- **Limone**
  - minimal features can support many applications
- **EgoSpaces**
  - interest and context are highly individualized notions
- **ServiceLime**
  - adaptive, predictable, and continuous provisioning is of the essence
- **CAST**
  - spatial and temporal operations are needed in real applications
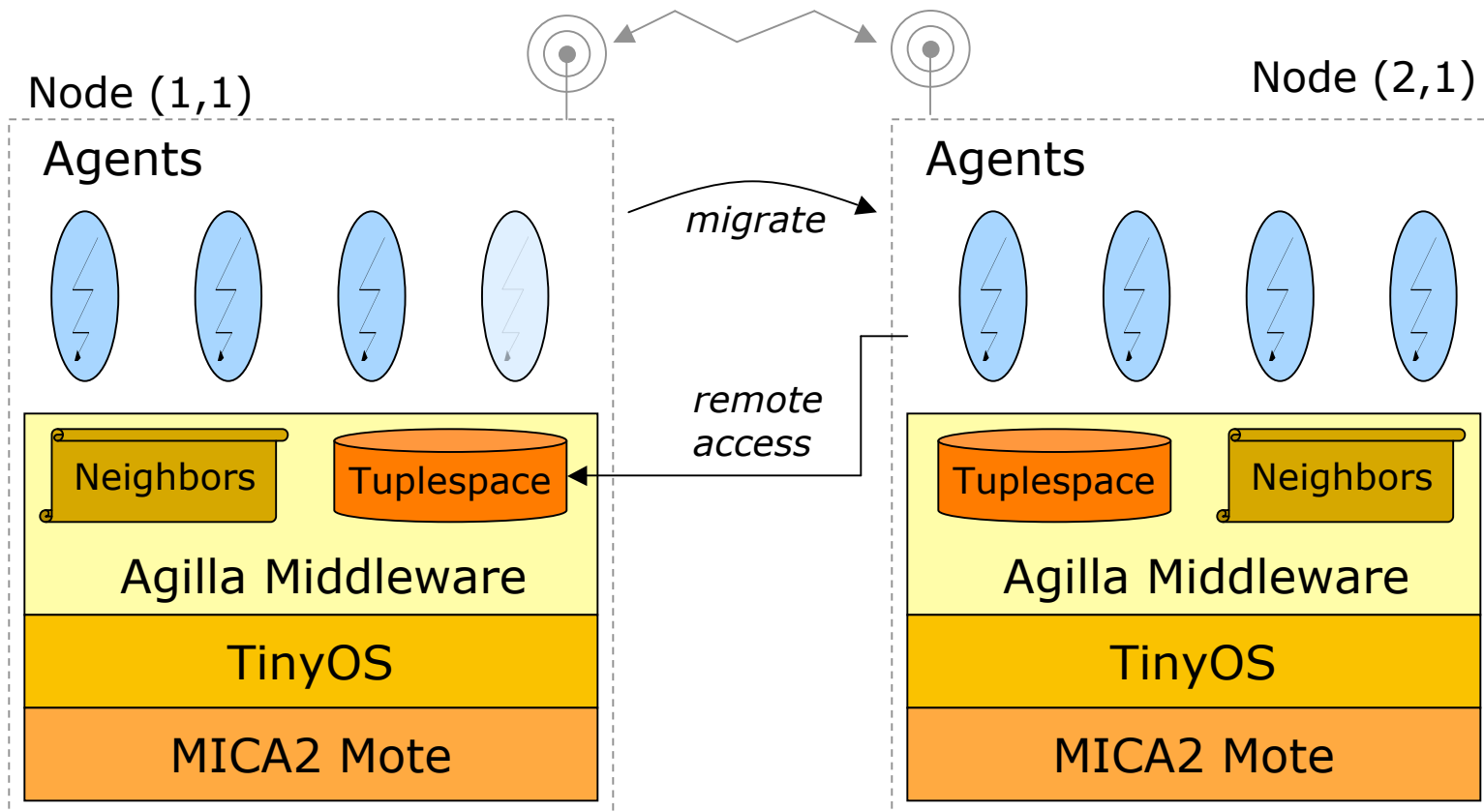
Coexistence

… in which we find out that needs are not always what they seem, wanting milk may be a veiled request for cream
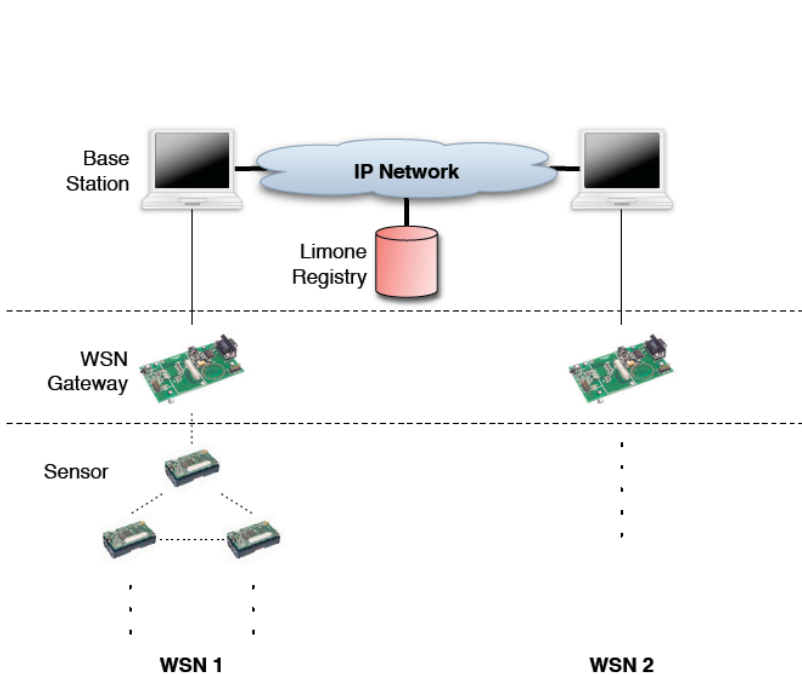
Sensor Networks

- A permanent and pervasive network
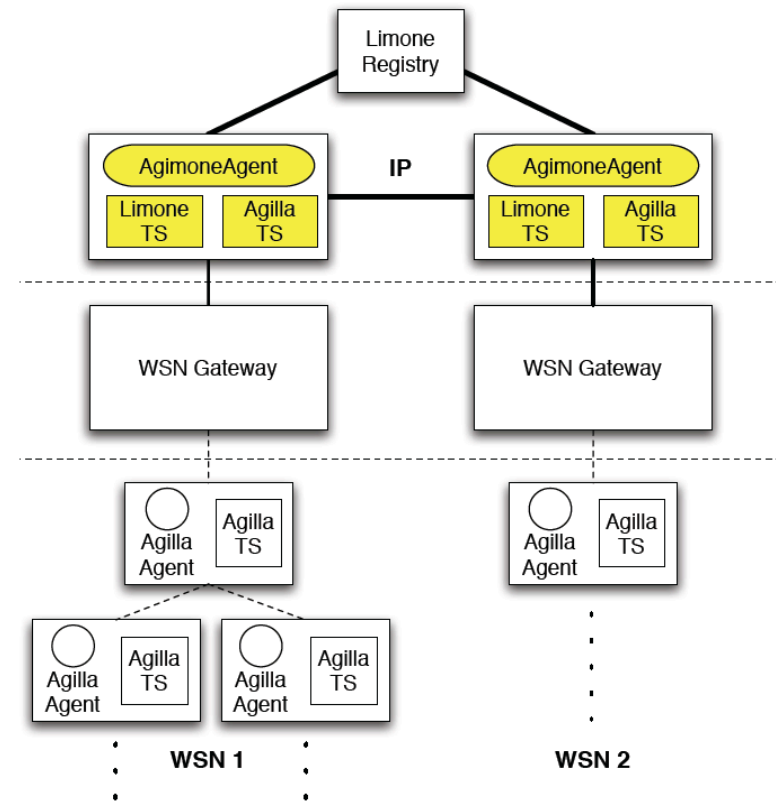- A shared computing resource

Agilla System Architecture

Node (1,1)

Node (2,1)

Agents

Agents

migrate

remote access

Neighbors

Tuplespace

Tuplespace

Neighbors

Agilla Middleware

Agilla Middleware

TinyOS

TinyOS

MICA2 Mote

MICA2 Mote

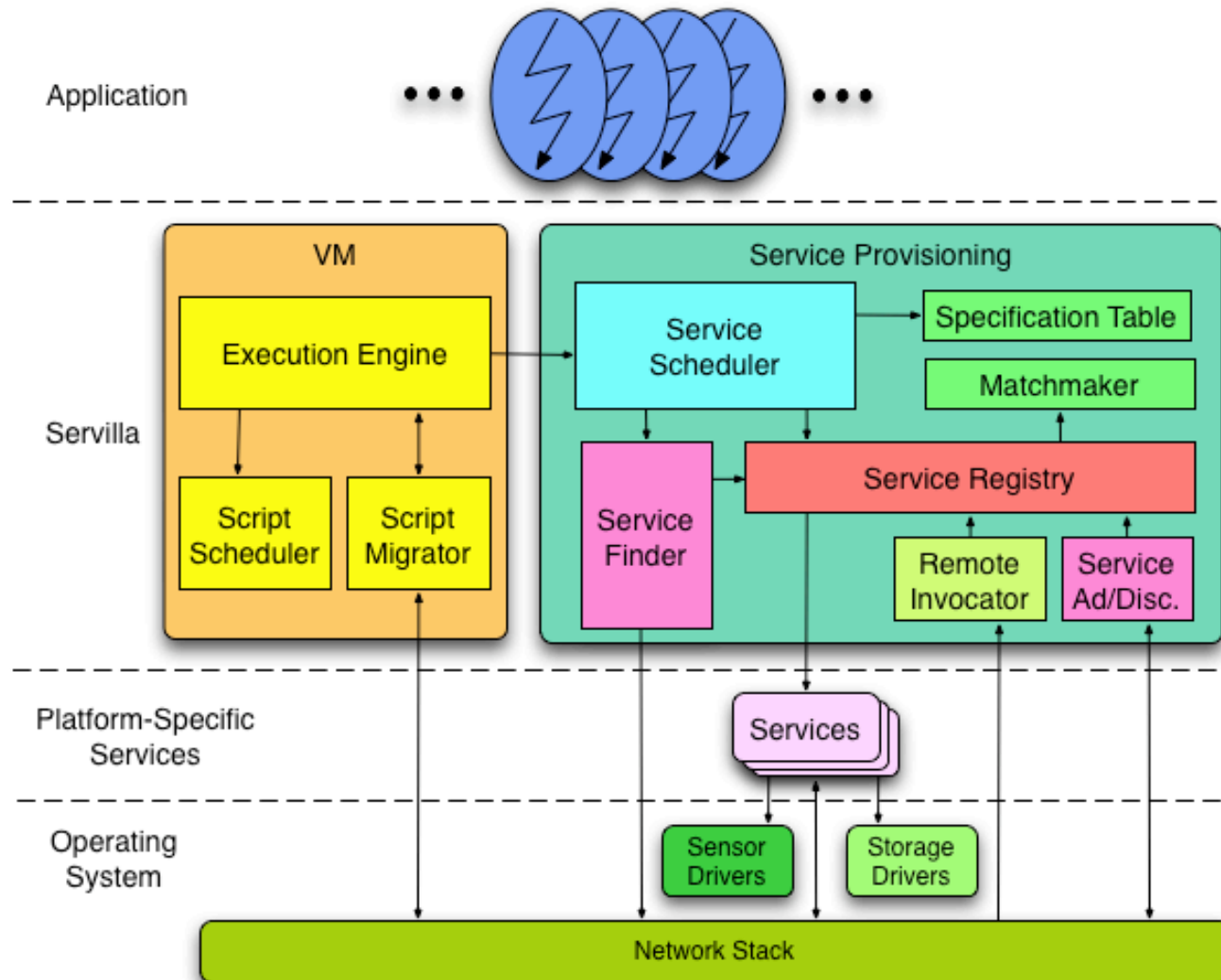# Spanning the Internet: Agimone

Network Architecture

System Components

# Coping with Heterogeneity: Servilla

Lessons Learned

- Agent technology is feasible and effective for the development of sensor network applications

- High level programming is necessary

- A flexible virtual machine offers significant gains

- Deployment of multiple applications requires both admission control and resource allocation

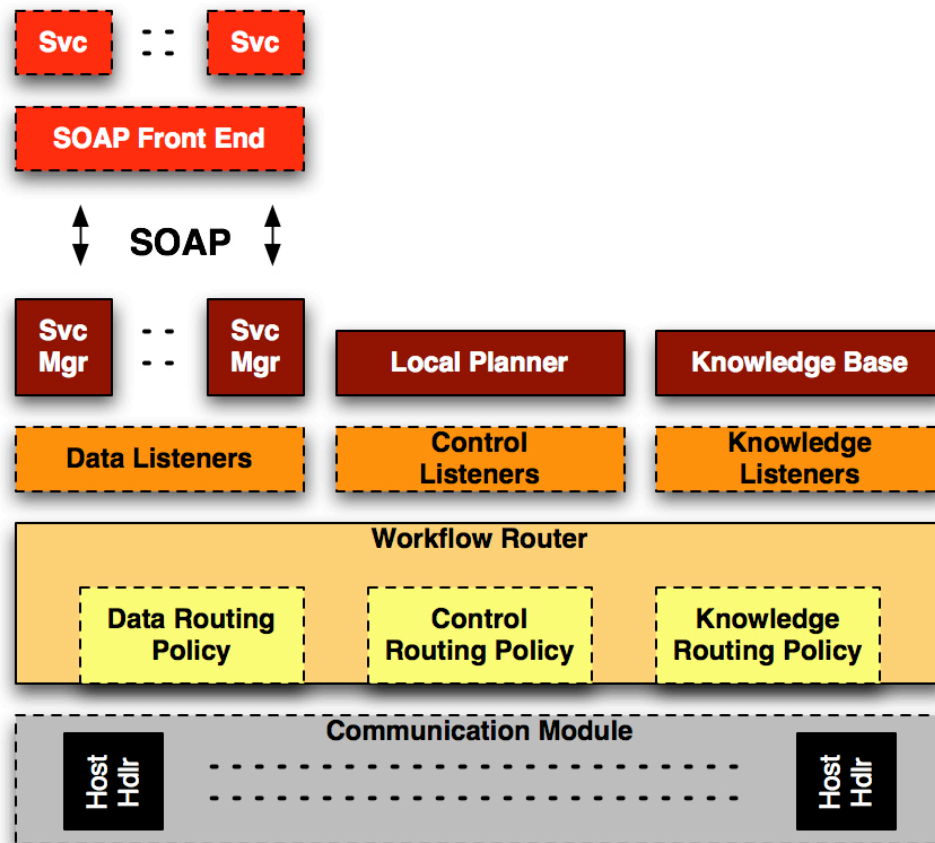- Heterogeneous resources demand increased reliance on dynamically bound services

Collaboration

… in which the internet is left behind and the work flows without

New Workflow Scenarios
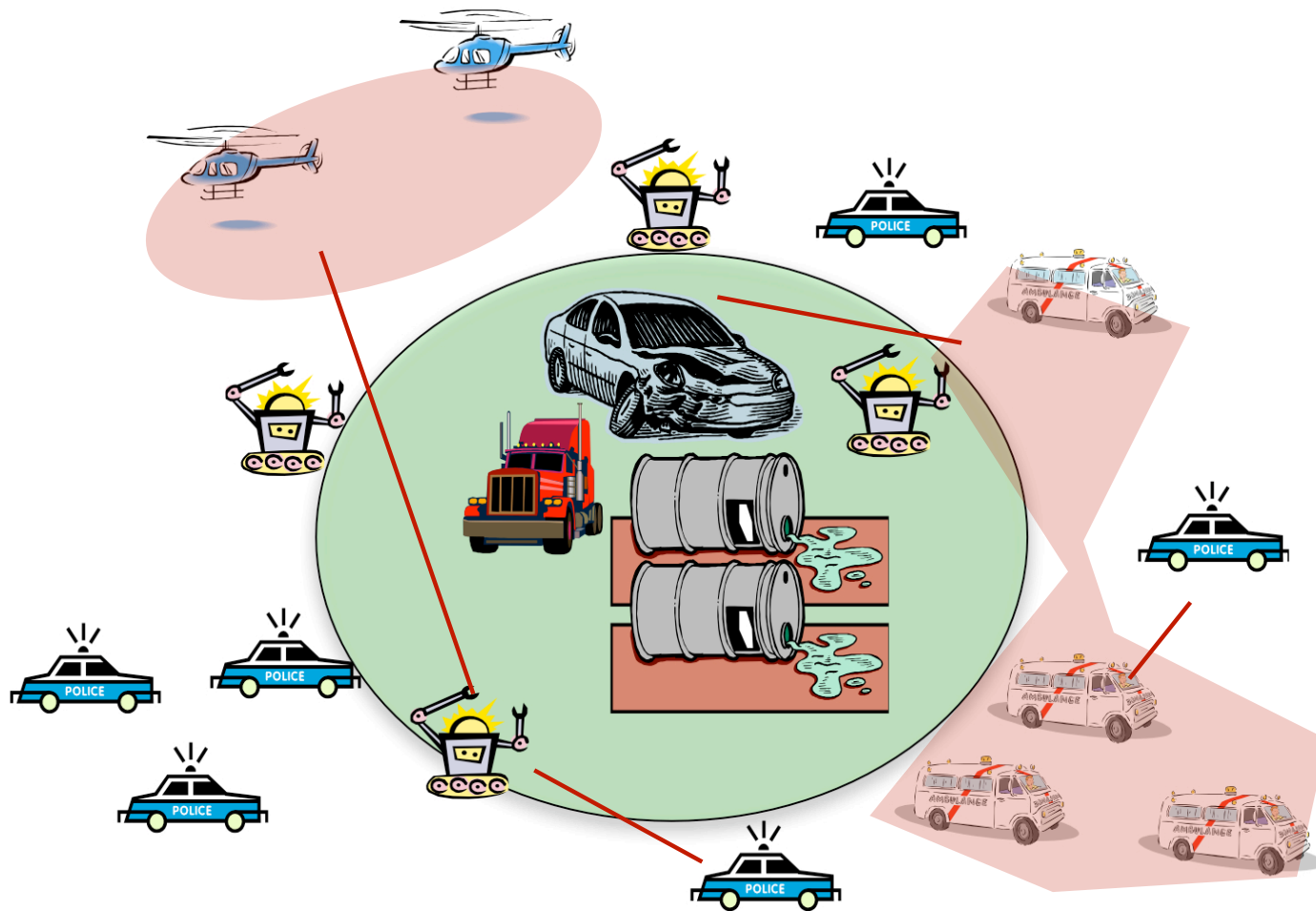
CiAN Engine

Lessons Learned

- Workflows involving the physical world entail spatiotemporal considerations

- Task allocation in mobile settings is a complex undertaking

- Situation awareness is an important planning ingredient (e.g., motion profiles)

- Corporate knowledge enhances the ability to get the job done

- Open and emergent workflows are promising new directions for collaboration in the real world

Conclusions

- **A new world order is emerging**
  - virtual and physical
  - personal and social
- **Mobility is integral to this fluid world which**
  - demands malleable and flexible applications
  - supported by new conceptual frameworks
  - made possible by middleware
  - rooted in new technical foundations

Thank you …

… and here is my address

http://www.cs.wustl.edu/~roman/