A scalable workbench for implementing and evaluating distributed applications in mobile ad-hoc networks

Johannes K. Lehnert, **Daniel Görgen**, Hannes Frey, Peter Sturm
System software and distributed systems
University of Trier
Germany
Mobile multi-hop ad-hoc networks

- Metropolitan sized networking
- Mobile devices
  - Wireless communication facilities
  - Localized location computation
- Direct communication only within transmission range
- Unpredictable network topology changes due to mobility
  - Network partitions
  - Permanent link failures
Mobile multi-hop ad-hoc networks

- Metropolitan sized networking
- Mobile devices
  - Wireless communication facilities
  - Localized location computation
- Direct communication only within transmission range
- Unpredictable network topology changes due to mobility
  - Network partitions
  - Permanent link failures
Application development in mobile multi-hop ad-hoc networks

- Challenging area
  - State-of-the-art still an open question
  - Self-organization
  - Small devices with many limitations

- Field trials expensive
  - Time, money, hardware, people
  - Critical mass needed for serious tests

- Uniform workbench
  - Develop and test in simulation first
  - Evaluate application in emulation
  - Use the *same* code in field trials
Marketplace communication

- Fixed geographic regions
  - High device density
  - Known position
- Devices at market acting on behalf of a user
- Geographic routing of agents/data
  - To and from marketplaces
- Negotiation at a marketplace
  - Geographic limited broadcast
  - Topology-based Routing
- Definition of home zones
  - Negotiation results are sent back to a defined home zone
Case Study: UbiBay

- Developed using workbench & proposed development process
  - Simulation
  - Emulation
  - Field-Trials

- Auction at marketplaces
  - Intended for low value goods
  - Direct neighborhood

- Agents
  - Auction agent
    - Controls the auction
  - Discovery agent
    - Discovers all auctions at marketplace
  - Bid agent
    - Bids on behalf of a user
Case Study: UbiBay

- Auction at marketplaces
  - Intended for low value goods
  - Direct neighborhood

- Agents
  - Auction agent
    - Controls the auction
  - Discovery agent
    - Discovers all auctions at marketplace
  - Bid agent
    - Bids on behalf of a user
Workbench: Simulation

- Scalable
  - 10000 devices possible
  - Precomputation for mobility and connectivity

- Focus on topological properties

- Extensible
  - Components defined as interfaces
  - Many default implementations (mobility, connectivity, network)

“Faster than real-time”
Workbench: Simulation II

- Intuitive, high abstraction level
  - Register as listener for neighbor discovery
  - Network messages = Java objects

- Code reuse

- Powerful Visualization
  - Freely definable
  - Multiple output targets: Swing/Java2D, OpenGL, PostScript, ...

“Concentrate on development, not on the simulator!”
Workbench: Hybrid mode

- Simulate network and devices
- Connect workstations or other devices to simulation
  - Replace simulated user behavior with GUI
  - RMI server controls simulation kernel
  - Mix of simulated and real user behavior possible
- Valuable for debugging
- “Get a feeling for the application”
Workbench: Real hardware

- Execution environment identical to simulation
  - Multiple threads, synchronization queues
  - Network implementation: WLAN + UDP unicast/broadcast
  - Positioning: GPS receivers
  - Neighbor discovery: periodic broadcasts
  - GUI: reused from hybrid mode

- Current implementation: PocketPC with IBM J9 VM
Summary

- Workbench approach works
  - Scalable:
    simulate thousands of devices in real-time
  - Intuitive and productive programming environment
  - Code reuse very effective

- It’s not finished:
  - Provide more mobility models
  - “Realistic” network model
  - Allow feedback from visualization