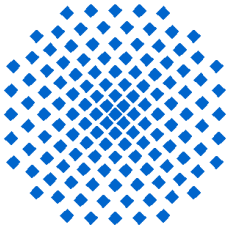


Universität Stuttgart

Institute of Parallel and
Distributed Systems (IPVS)

Universitätsstraße 38
D-70569 Stuttgart

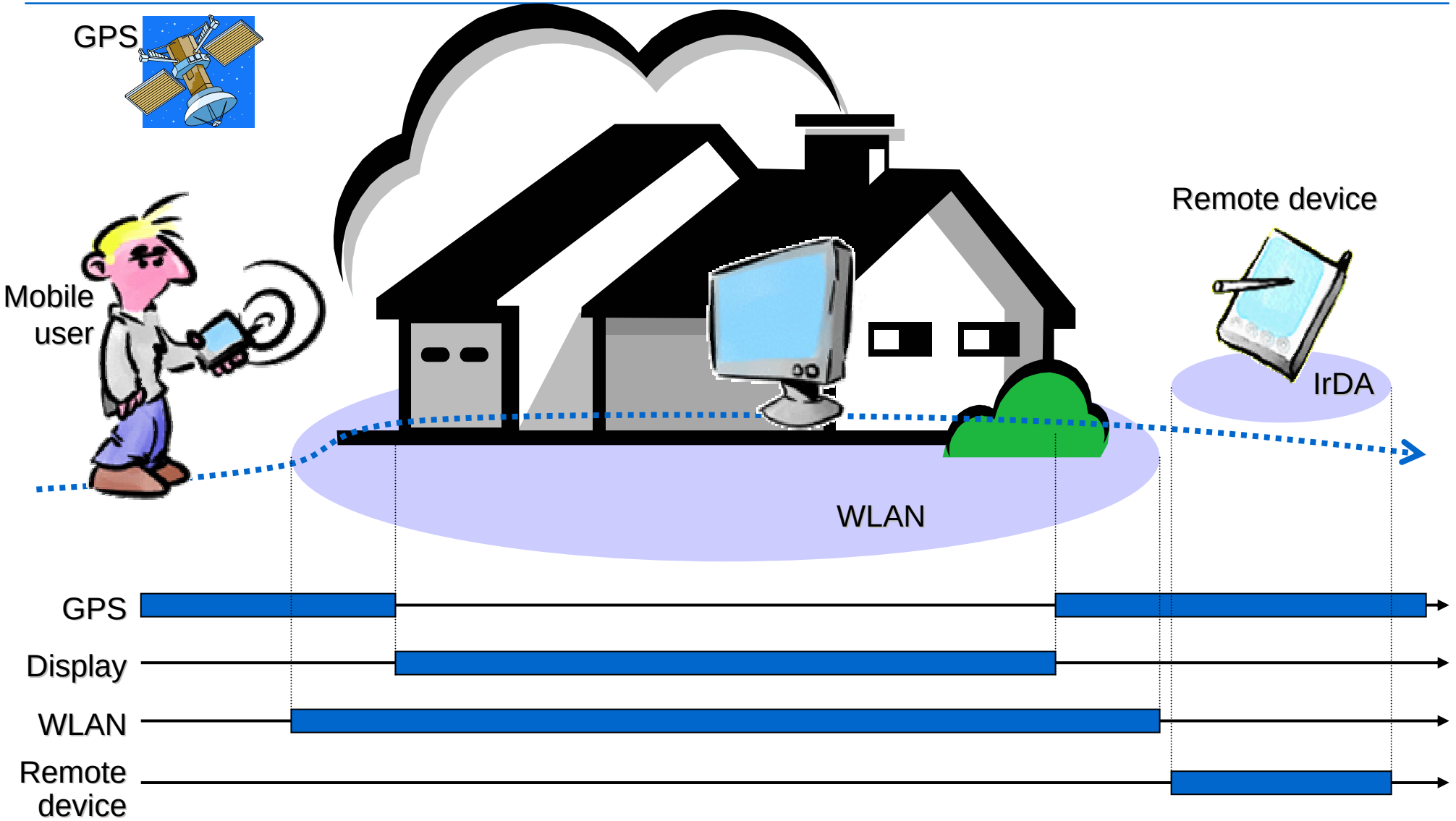
Energy-Efficient Cluster-based Service Discovery for Ubiquitous Computing

Gregor Schiele

Christian Becker

Kurt Rothermel

Ubiquitous Computing: A Scenario



Ubiquitous Computing: The Need for Adaptation

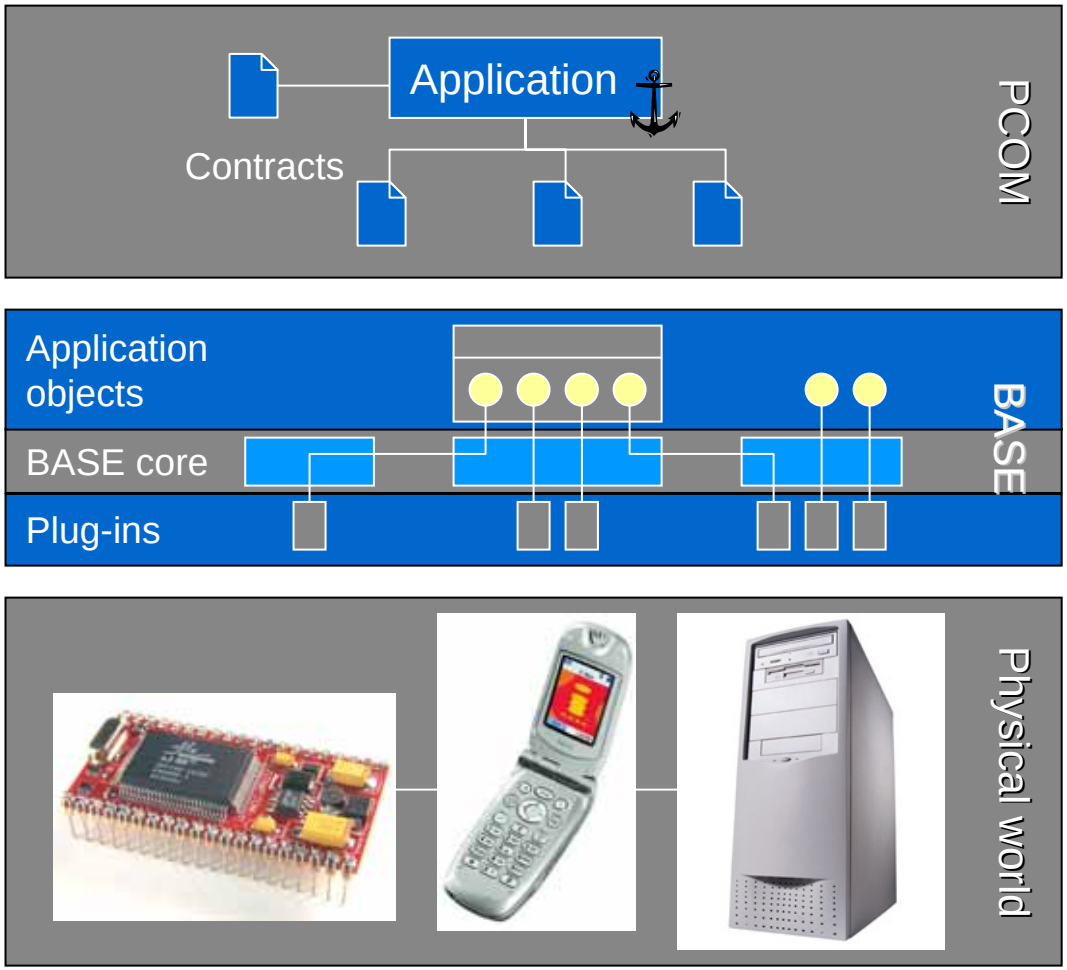
- **Ubiquitous Computing (UC) characteristics:**
  - Many interconnected devices
  - Mobile and battery-operated
  - Distributed applications
  - Highly dynamic execution environments

→Applications must adapt (at runtime) to fluctuating availability of

- **devices and**
- **services**

The Peer-to-Peer Pervasive Computing Project

- **Goal:**
  - System support for adaptive applications in UC
- **3PC enables dynamic adaptation on communication (BASE) and application (PCOM) layer**
- **Current Research Areas:**
  - Automated Application Configuration
  - Secure Communication
  - Energy-Efficient Service Discovery

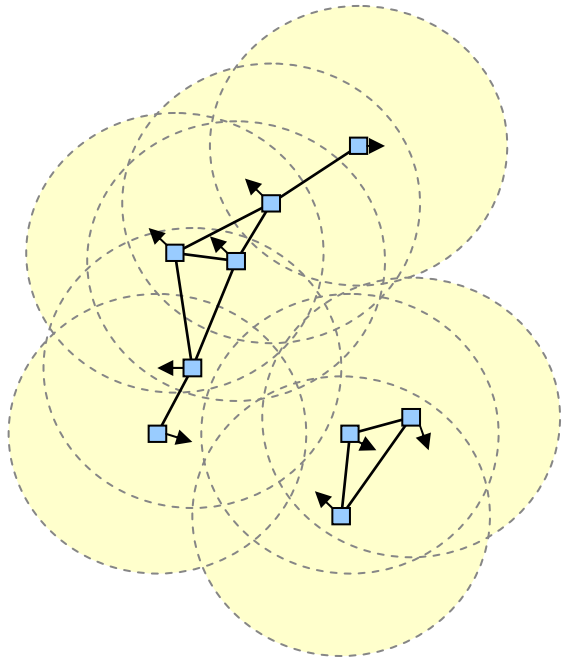


Contents

- System Model
- Requirements
- Discovery Approaches
- SANDMAN
- Evaluation
- Conclusion and Future Work

System Model

- **Nodes:**
  - Mobile or stationary
  - Operated on limited battery power
  - Two modes of operation:
    - *Sleeping*: no calculations or communication
    - *Activated*: node fully operational
  - Node state resilient to sleeping
- **Network:**
  - Multi-hop MANET
  - Local broadcast ability
  - Reliable unicast, unreliable multicast



Service Discovery Requirements

- **Energy-efficiency**
  - Must enable energy-efficient node operation

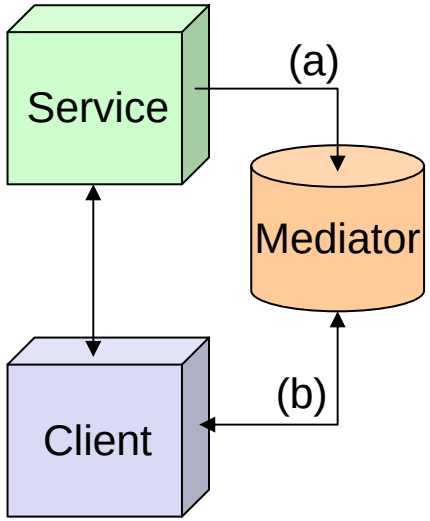
  →Our approach: Maximize sleep times of idle nodes
- **Low discovery latency**
  - Must enable prompt discovery of new devices and their services

  →Minimize delay due to sleeping nodes
- **Decentralized operation**
  - Must work despite frequent and unpredictable network partitions

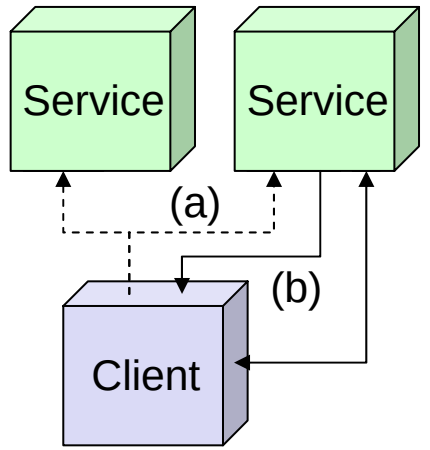
Discovery Approaches

- **Mediator-based Discovery (e.g. Jini, INS, SDS)**
  - Services register at mediator (a)
  - Clients lookup services there (b)
  - No discovery if mediator not reachable

→Not appropriate for decentralized environments



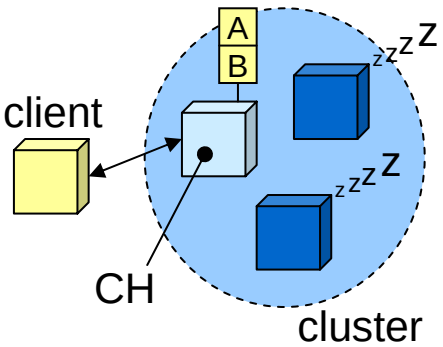
- **Peer-based Discovery (e.g. UPnP)**
  - Clients multicast requests (a)
  - Suitable services respond (b)
  - Or: Services multicast announcements periodically
  - Inefficient (communication, sleeping nodes)



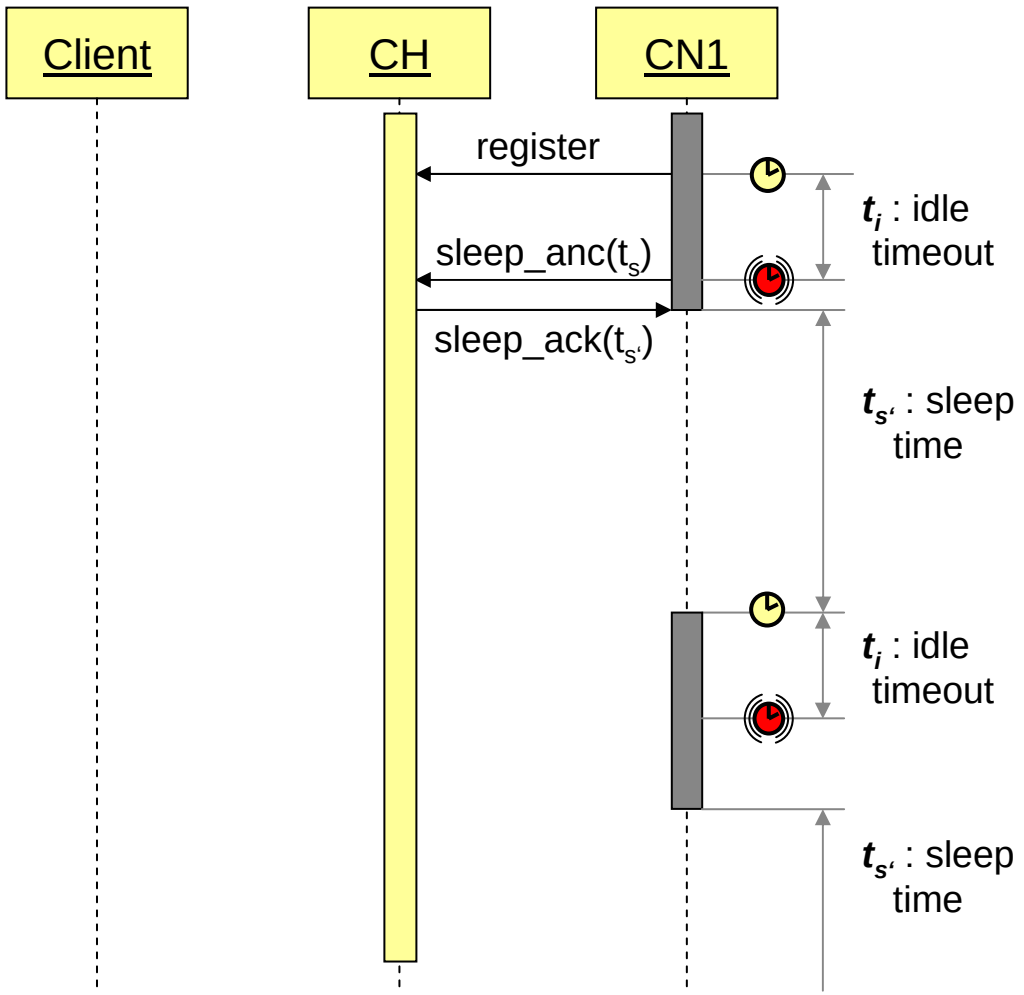
SANDMAN: Overview

- Service Awareness aNd Discovery in Mobile Ad-hoc Networks

- **Idea:** Combine mediator-based and peer-based discovery
- **Overall approach:**
  - All nodes start with peer-based discovery
  - Organize MANET dynamically into node clusters
    - 1 cluster head (CH) per cluster
    - n clustered nodes (CNs) per CH
  - CNs sleep if idle
  - CHs stay awake and act as mediators for lookup requests from clients

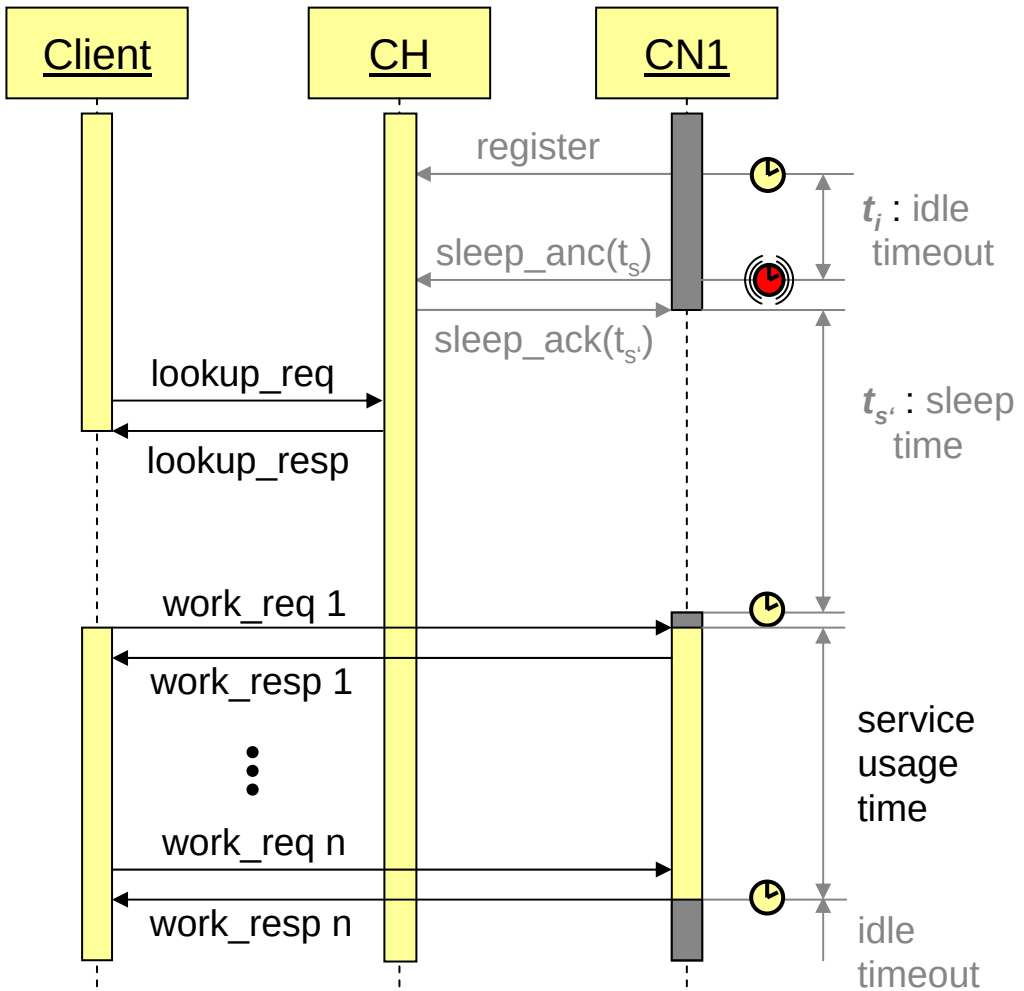


SANDMAN: Service / Cluster Head Interaction



- Assumption: cluster formed, CH elected
- At cluster entry: CN registers its services at CH (*register*)
- After registration CN starts idle timer t_i
- If timeout occurs: CN initiates sleeping (*sleep_anc*)
- CH acknowledges sleeping (*sleep_ack*)
- Actual sleeping time t_s negotiated between CH and CN
- When CN awakes it restarts t_i

SANDMAN: Client / Cluster Head Interaction

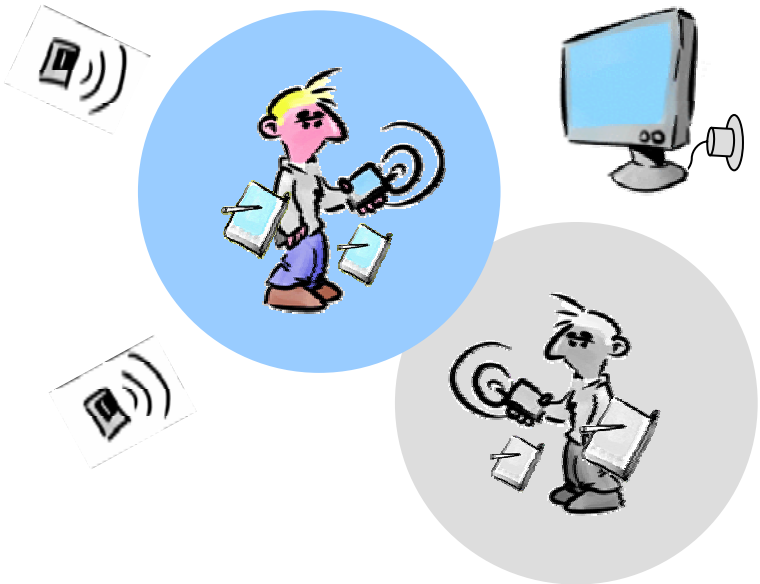


- To discover service, client multicasts lookup request (*lookup_req*) to CHs
- CH responds with service list (*lookup_resp*)
  - Service description
  - Service provider (CN)
  - Time until wakeup
- Client
  - Selects ‘best’ service
  - Waits until provider is awake
  - Contacts it

SANDMAN: Cluster Management

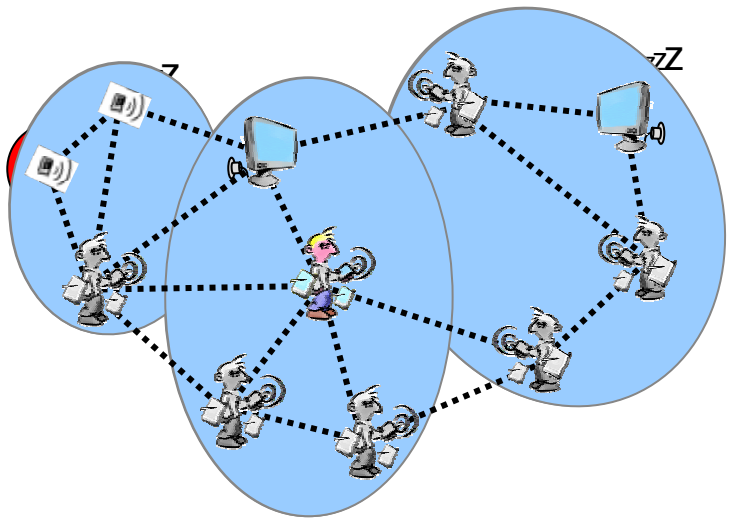
- **Basic Clustering (PAN)**
  - Group PAN devices
  - PAN devices identified by user
    - Authentication, SmartIts friends

  →Group predetermined



- **Extended Clustering (MANET)**
  - Group arbitrary devices

  →Connectivity must be regarded when deactivating nodes



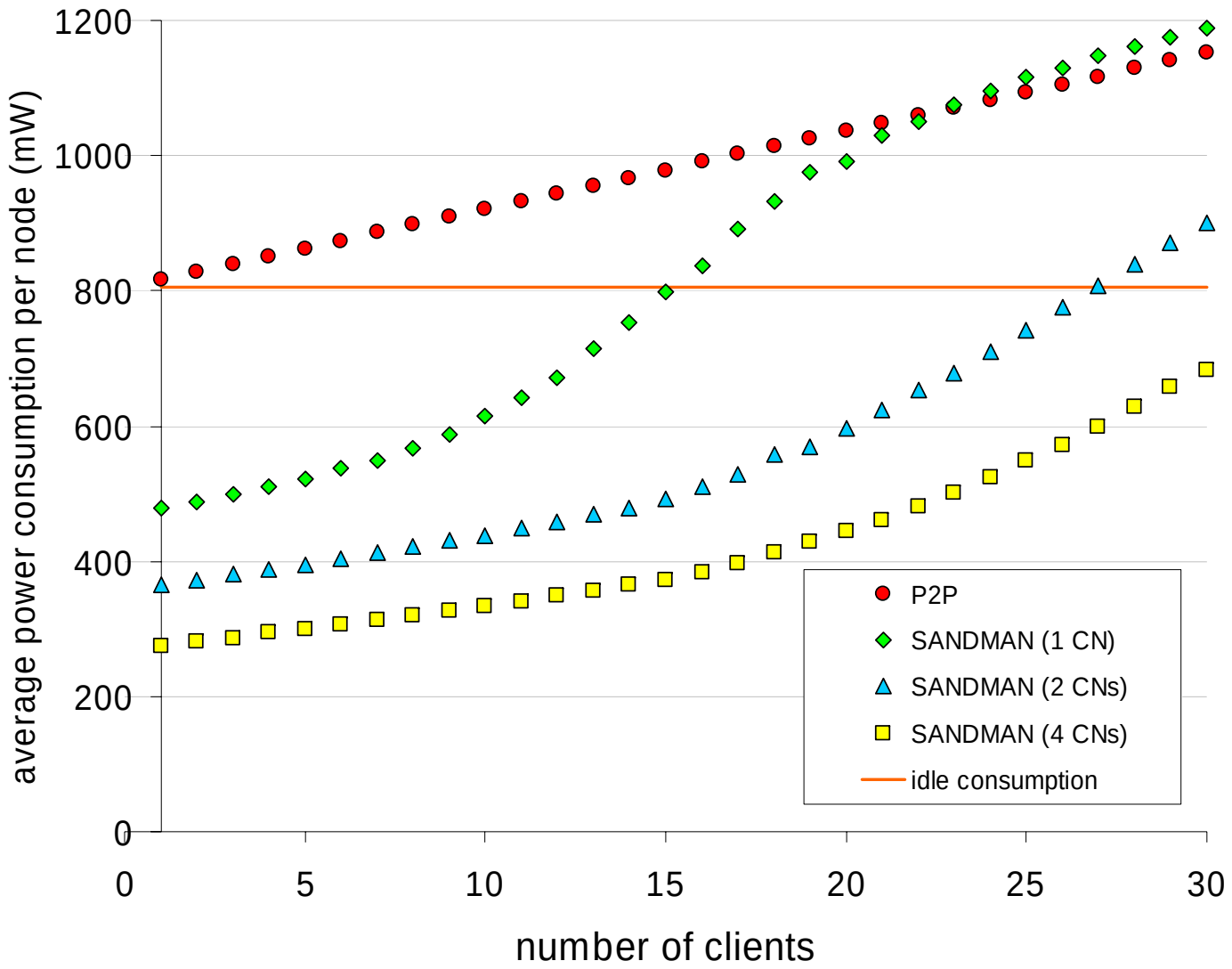
Evaluation: Simulation Scenario

- Simulations conducted using network simulator NS2
- **Cluster:** fixed cluster with 1 CH (cluster management omitted)
- **Services:** 1, 2, 4 CNs, t_i=1s, t_s=9s
- **Clients:** 1-30, random discovery interval [0, 8] s of randomly selected service
- **Measured parameters:**
  - Energy consumption
  - Discovery & interaction latency
- **For comparison:** peer-based approach without energy saving (P2P)

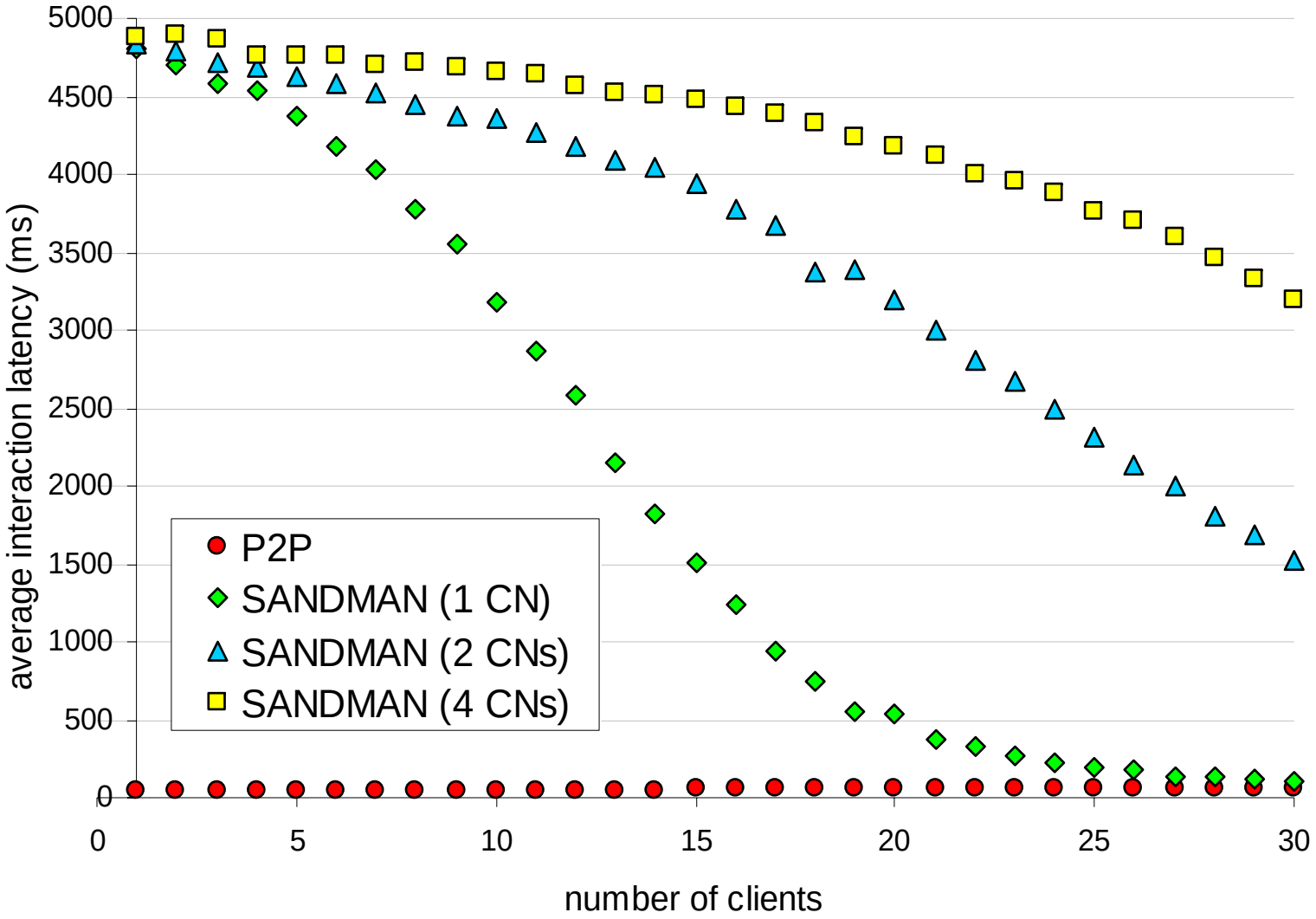
Orinoco PC Gold	
Sleep:	60 mW
Idle:	805 mW
Recv:	950 mW
Send:	1400 mW

(Shih et al. MobiCom02)

Evaluation: Average Power Consumption



Evaluation: Average Interaction Latency



Conclusion

- SANDMAN provides energy efficient service discovery in UC
- Readily useable with today's hardware
- Cluster-based organization
  - CHs stay awake permanently, answer discovery requests
  - CNs sleep if idle
  - CNs answer service requests in wakeup mode

→High energy savings achievable, discovery delays unaffected, interaction delays can be high

Future Work

- **Current Work:**
  - Implementation and integration in BASE
  - Enhanced cluster management (free clustering)
- **Next Steps:**
  - Integration of multiple communication interfaces
  - Evaluation using Network Emulation Testbed (NET)

Further Information

- **On the Web:**
  - **3PC:** http://www.3pc.info
  - **SANDMAN:** http://www.3pc.info/sandman

- **By Email:**
  - **Gregor Schiele:** gregor.schiele@informatik.uni-stuttgart.de