Mobility Management in Sensor Networks

Muneeb Ali†‡, Thiemo Voigt‡, and Zartash Uzmi†

†LUMS, Pakistan

‡SICS, Sweden

Outline

Two recent research trends that motivate our work:

(a) Towards a Sensor Network architecture

(b) Mobility in Sensor Networks

- Towards a Sensor Network Architecture
  - Internet vs Sensor Networks
  - Sensor-Net Protocol (SP)
- Mobility in Sensor Networks
- Mobility-Management in Sensor Networks
- On-going Work
- Open Issues
- Conclusion

Internet vs Sensor-Nets

The Internet

- Independent hosts
- End to end flows
- Two tier architecture
- Wired (generally)
- Latency
- Throughput
- Bandwidth is relatively cheap

Sensor Networks

- Collaborative use
- Collect, disseminate, ...
- Ad-hoc (more homogeneous)
- Low power wireless
- Wake time
- Very low utilization
- Bandwidth is expensive

Reference: Philip Lewis, ICSI Talk, May 2004

Internet vs Sensor-Nets

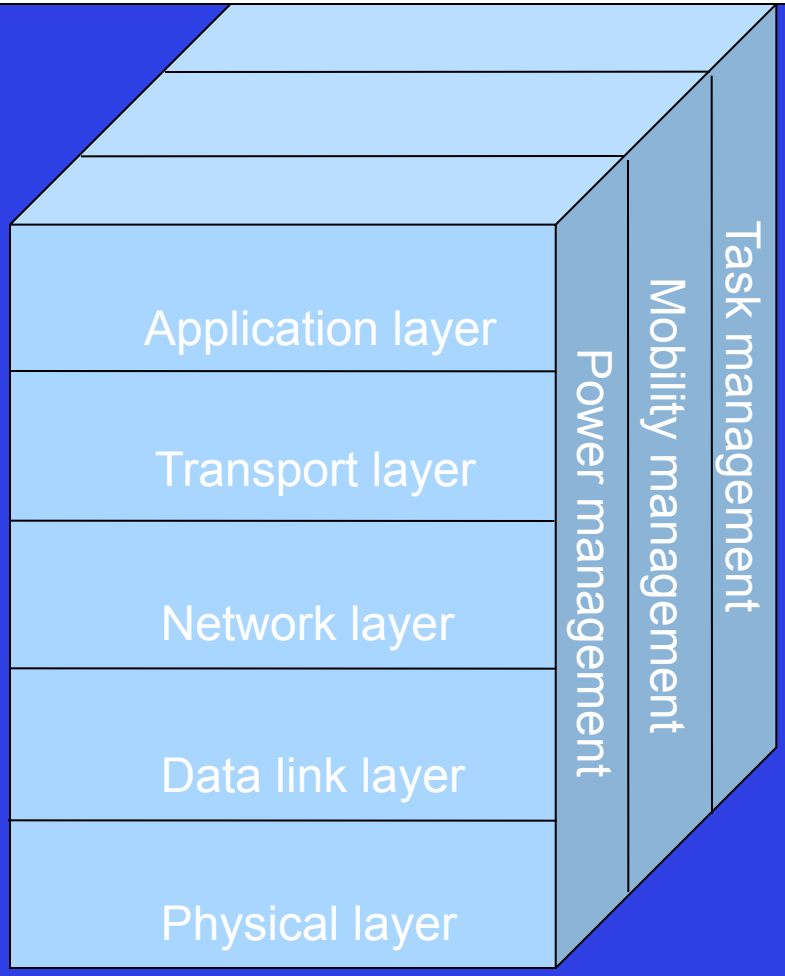
Lessons Learned

- Internet solutions generally do not apply apply to sensor networks
- Their underlying techniques do
- Apply, change and adapt to the peculiarities of sensor networks

Reference: Philip Lewis, ICSI Talk, May 2004

Towards a Sensor-Net Architecture

Traditional view of the sensor network protocol stack
(not strictly enforced)



Reference: Ian Akyildiz et al., Survey Paper, IEEE ComMag, Aug 2002

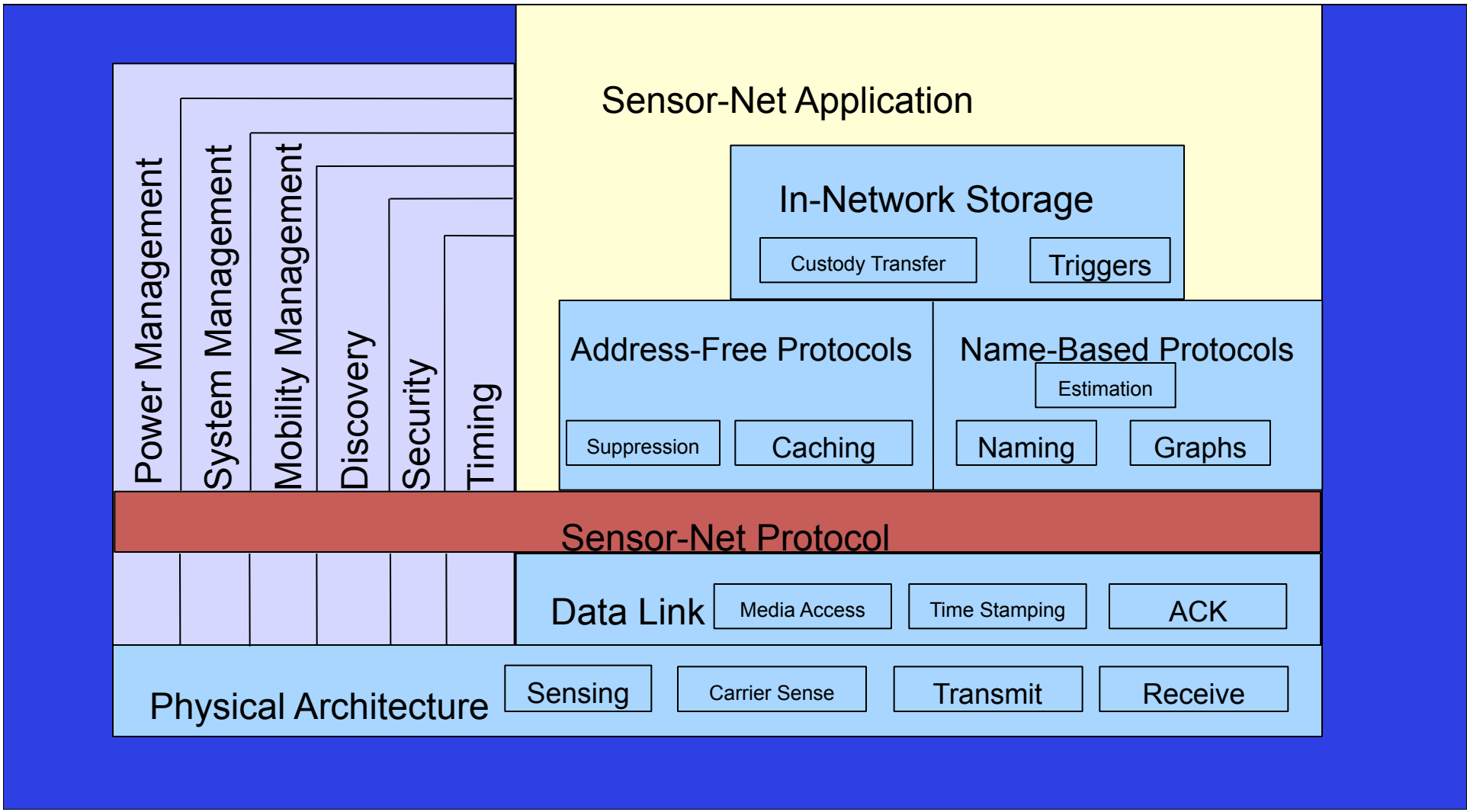
Towards a Sensor-Net Architecture

- Alphabet soup of protocols and subsystems
- Widely differing assumptions about:
  - the rest of the system and,
  - how its part should interact
- Vertically integrated designs
  - work with own set of components
  - unable to inter-operate
- No standards that the protocols and solutions need to conform to
  - good for research
  - bad for interoperability

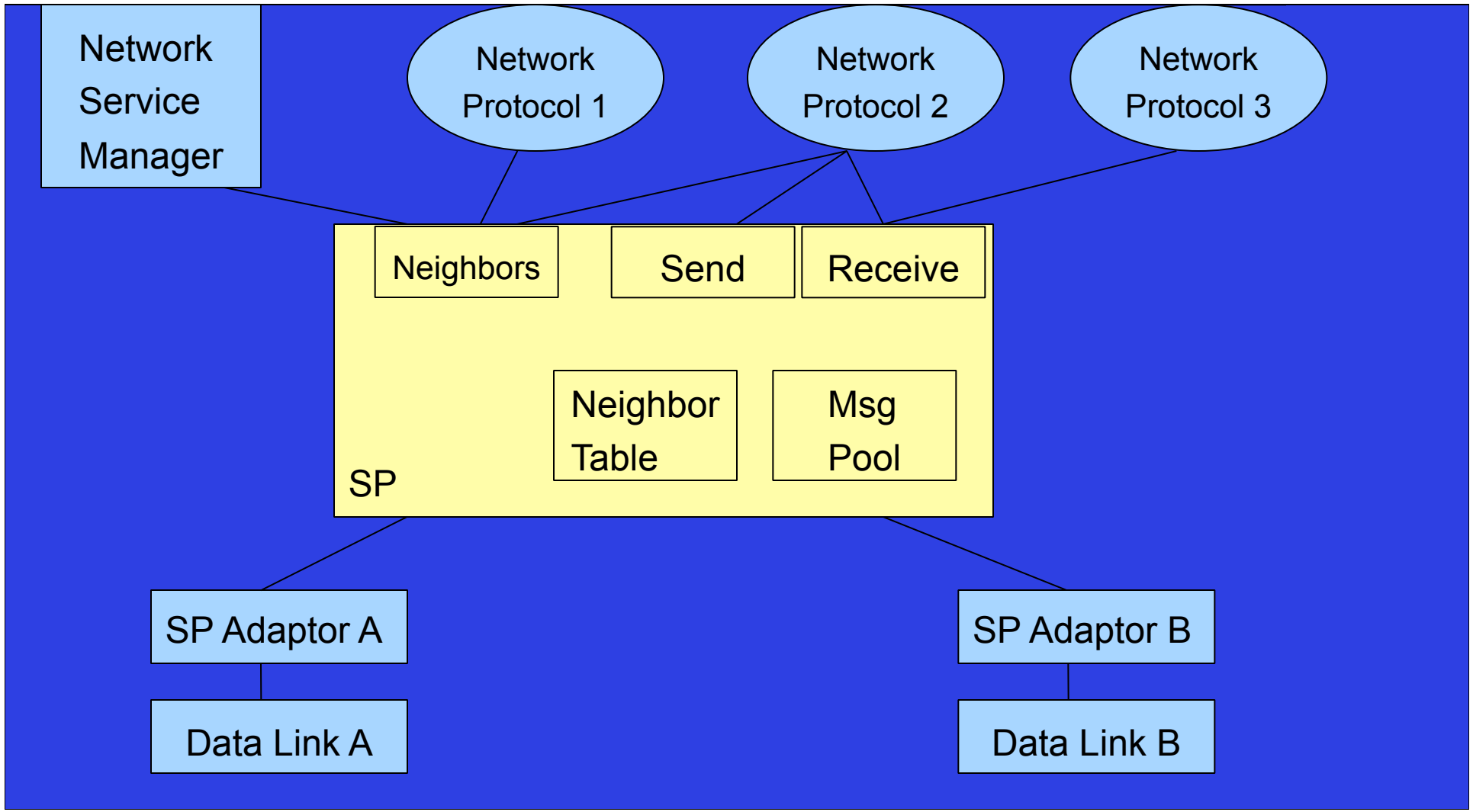
Sensor-Net (SP) Protocol

- Only of the early encouraging steps towards a sensor-net architecture
- Unlike IP, SP sits between the network layer and the data link layer
  REASON: processing potentially occurs at each hop not just at end points
- Allows multiple network protocols and link technologies to co-exist
- Abstraction could be implemented in any OS
- SP performs three main operations:
  a) Data SEND
  b) Data RECEIVE
  c) Neighbor Management
- Main differences from IP
  a) feedback e.g. Congestion, phase shift
  b) network protocols can request urgent/reliable service
  c) allow network and link layer to share link information

Towards a Sensor-Net Architecture



Sensor-Net (SP) Protocol



SP vs ZigBee

Apart from SP there are other emerging standards as well e.g. ZigBee

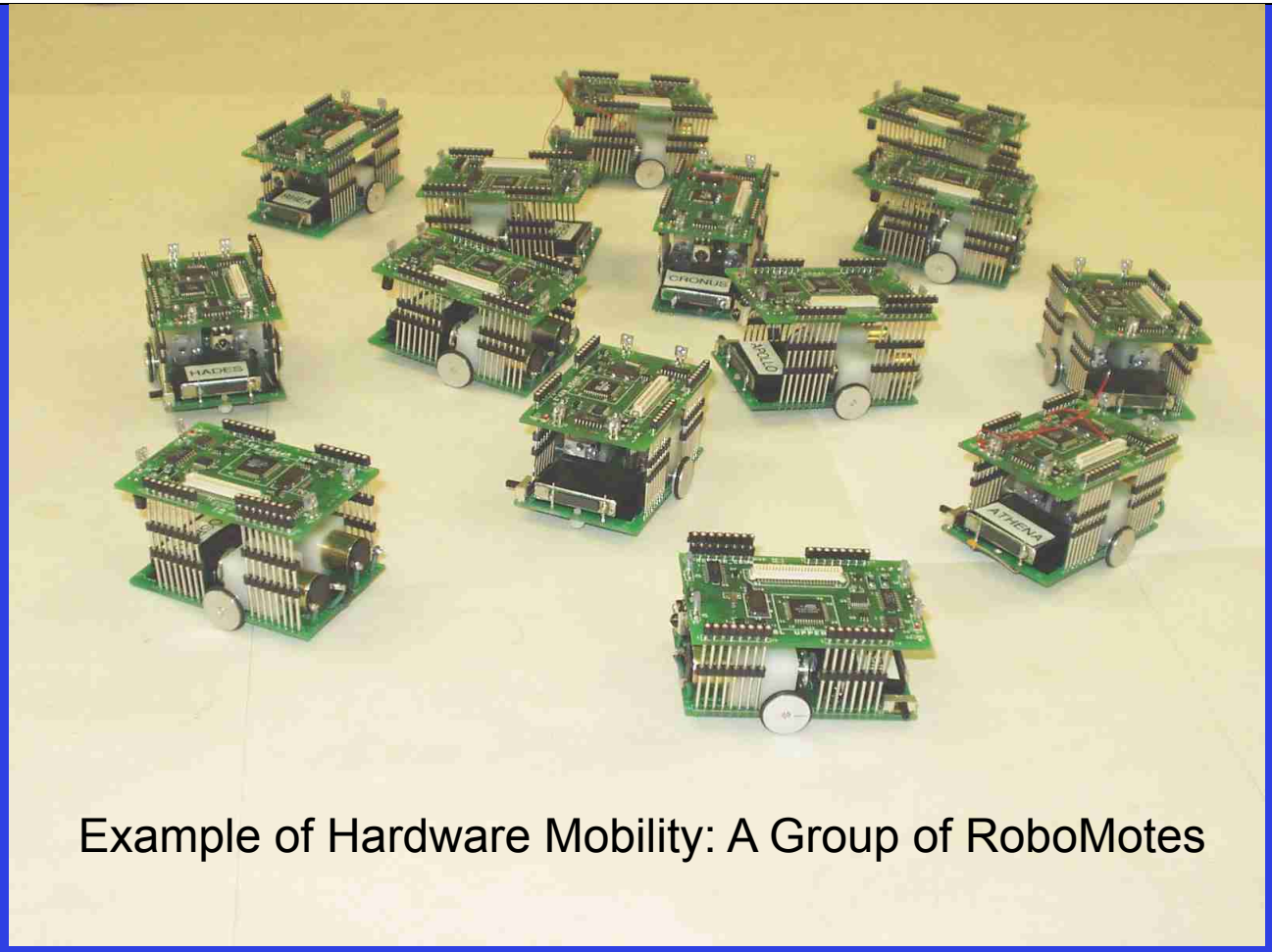
"ZigBee proposes a classic layered architecture, but each layer assumes a specific instance of the surrounding layers: e.g., the routing layer assumes the IEEE 802.15.4 link and physical layers. An architecture build on static technologies is destined for obsolescence"

Reference: Joe Polastre et al., "A Unifying Link Abstraction for Wireless Sensor Networks", In Proc. ACM SenSys 2005.

Mobility in Sensor Networks

- Research community generally ignores mobility in sensor networks
  - they assume static sensor nodes
- Recent works have enabled mobility in sensor-nets
  - e.g. RoboMote [Ref: K. Dantu et al., RoboMote paper, IPSN 2005],
  - and Parasitic Mobility [Ref: MIT Media Lab, Parasitic Mobility paper, Pervasive 2005]
- Medical care or disaster response applications use mobile sensor nodes
  - e.g. sensors attached to doctors or first responders
- Most protocols designed for static sensor networks perform poorly in mobile scenarios
  - e.g. MAC protocols [Ref: M. Ali et al. MMAC paper, IEEE IPCCC 2005]
- Mobility could even improve other things like:
  - coverage [Ref: B. Lie et al., Mobility Improves Coverage of Sensor Networks, Mobihoc 2005]
  - localization [Ref: David Evans et al., Localization for Mobile Sensor Networks, Mobicom 2004]

Mobility in Sensor Networks



Example of Hardware Mobility: A Group of RoboMotes

Image courtesy RobotMote – USC

Mobility in Sensor Networks

Example of Medical Care and Disaster Relief Applications

Image courtesy CodeBlue - Harvard

Mobility-Management

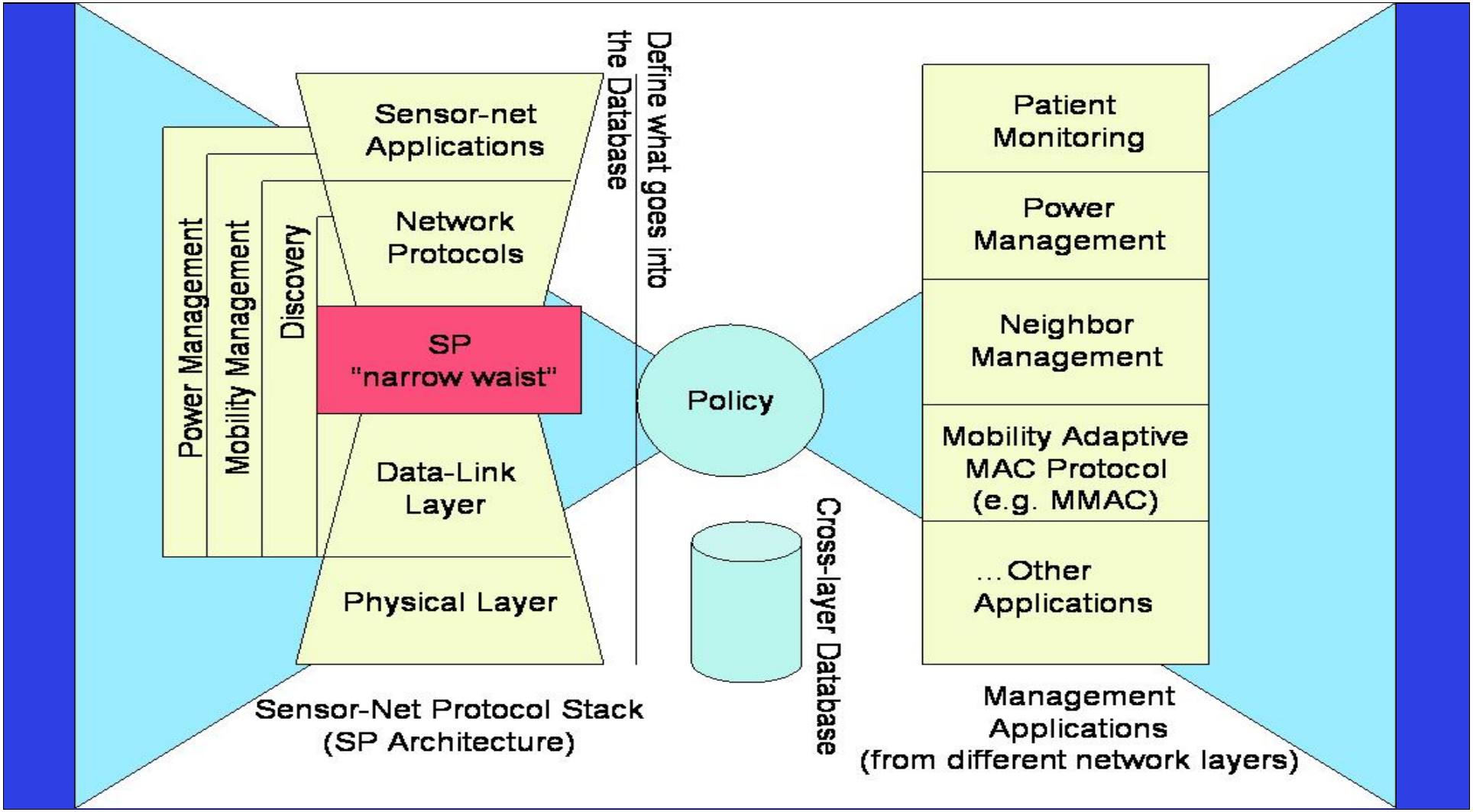
- Mobility information could be required:
  - at the application layer (e.g. monitoring physical movement of depression patients)
  - at the network layer (e.g. neighbour discovery, route maintenance)
  - at the MAC layer (e.g. MMAC: mobility adaptive MAC [IEEE IPCCC 2005])

- Protocols at different layers:
  - could gather, store and manage mobility information individually (current practice)
  - could make use of a cross-layer service that takes care of their mobility needs (our proposal)

- Instead of exporting information between different layers (redundant) it is more useful to:
  - import mobility information into a separate management database
  - make this database visible across all layers

- Standardizing what goes *into* the database:
  - enables network protocols and management applications to evolve independent of each other
  - helps in moving towards a sensor-net architecture

Mobility-Management



Mobility-Management

- Our “bow-tie” mobility management design:
  - does NOT take any stance on Time Synchronization (works with any)
  - does NOT take any stance on naming (but assumes that nodes have unique addresses)
- Cross-layer database is implemented as a shared buffer and:
  - is populated by information collected from the left-side of the bow-tie (SP network stack)
  - provides services to management applications (right-side of the bow-tie)
- For mobility estimation:
  - we propose to use AR-1 model [Ref: Z. Zaidi et al., Globecom 2004 and Secon 2004]
  - more accurate AR-3 model is too computationally intensive for sensor nodes
- Accuracy of mobility estimation depends on underlying localization mechanism
- There is some communication overhead to gather and update mobility information of nodes
  - is it worth it?

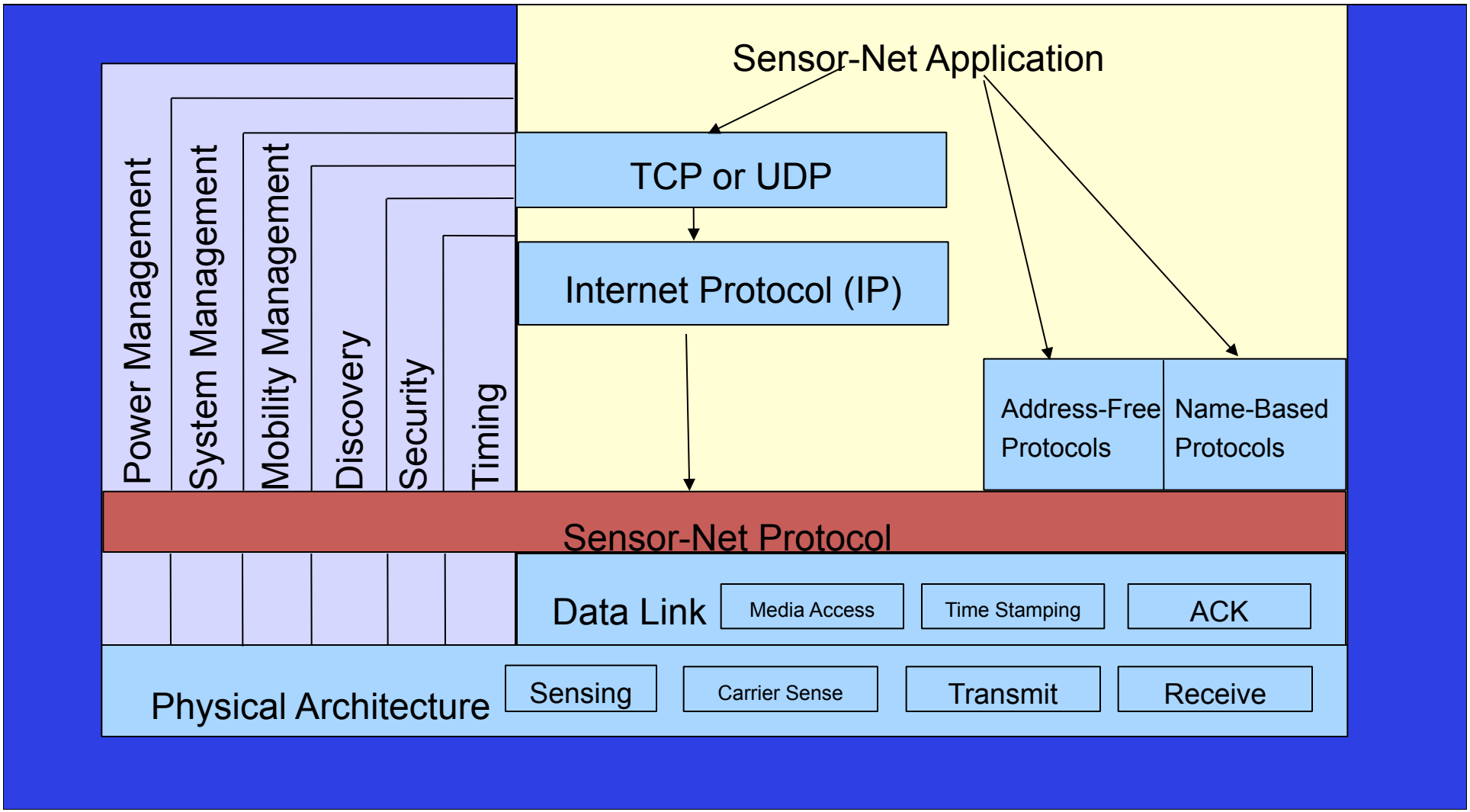
On-going Work

- Currently implementing SP and the mobility-management cross-layer service
  - on Contiki Operating System [Ref: A Dunkels et al., Contiki paper, EmNets-I 2004]
  - using Protothreads [Ref: A Dunkels et al., Protothreads paper, RealWSN 2005]
- For simulations:
  - using COOJA simulator for Contiki [Ref: F. Osterlind, SICS Tech. Rep. T2006-05]
  - using COOJA reduces the time to map simulation code to real deployments
- For mobility evaluations:
  - implementing realistic mobility models [Ref: T. Camp et al., WCMC 2002]
  - and using real mobility traces [Ref: D. Kotz et al., ACM MSWiM 2004]

Open Issues: Standard Database?

Node ID	Predicted (X,Y)	For Time	Original Time Stamp
7	(23,5)	T1 + i	T1
3	(102,17)	T2 + j	T2
15	(0,96)	T3 + i	T3
7	(24,6)	T1 + j	T1

Open Issues: IP over SP?



Conclusions

- Current sensor-net literature
  - presents an alphabet soup of protocols and sub-systems
  - which do not inter-operate and make varying assumptions about others
- SP is an encouraging step towards a sensor network architecture
- Researchers assume "static sensor nodes" – an assumption that might not be valid now
- SP's unifying link-abstraction and our mobility-management framework could:
  - provide efficient mobility handling
  - enable efforts from different research groups to inter-operate with each other
- Sensor-net community should make use of SP with mobility-management as a cross-layer service to provide a *standardized* yet *flexible* framework for future research

Further Information

Muneeb Ali

http://muneeb.org

Thank You !