

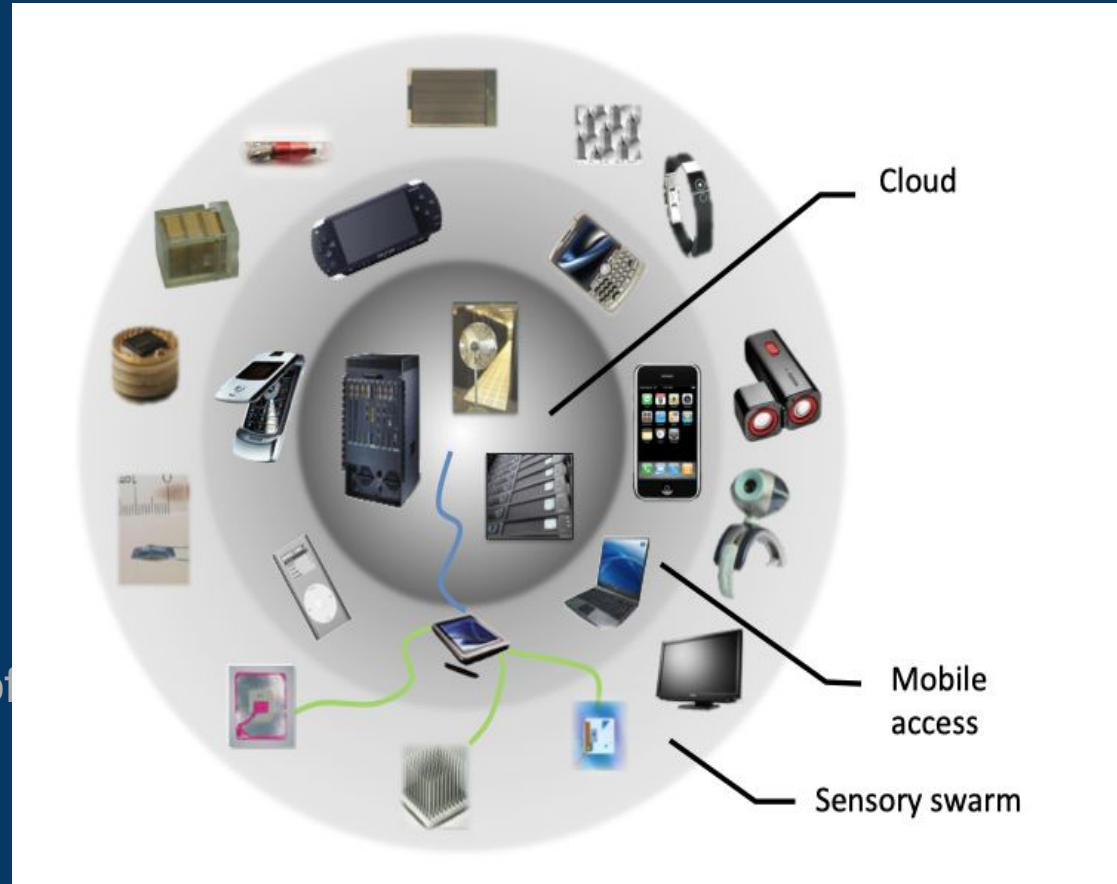
The Swarm at the Edge of the Cloud

Edward A. Lee, Jan Rabaey, David Blaauw, Prabal Dutta, Kevin Fu, Carlos Guestrin, Björn Hartmann, Roozbeh Jafari, Doug Jones, John Kubiawicz, Vijay Kumar, Rahul Mangharam, Richard M. Murray, George Pappas, Kris Pister, Anthony Rowe, Alberto Sangiovanni-Vincentelli, Sanjit A. Seshia, Tajana Simunic Rosing, Ben Taskar, John Wawrzynek, David Wessel

Presented by: Andrew Nguyen

Sensory Swarm

- Wirelessly interconnected and interact with the cyber-cloud
- Necessary is a three level model:
 - cloudbackbone
 - personal devices
 - swarm devices
- Something along the lines of an “app” is needed

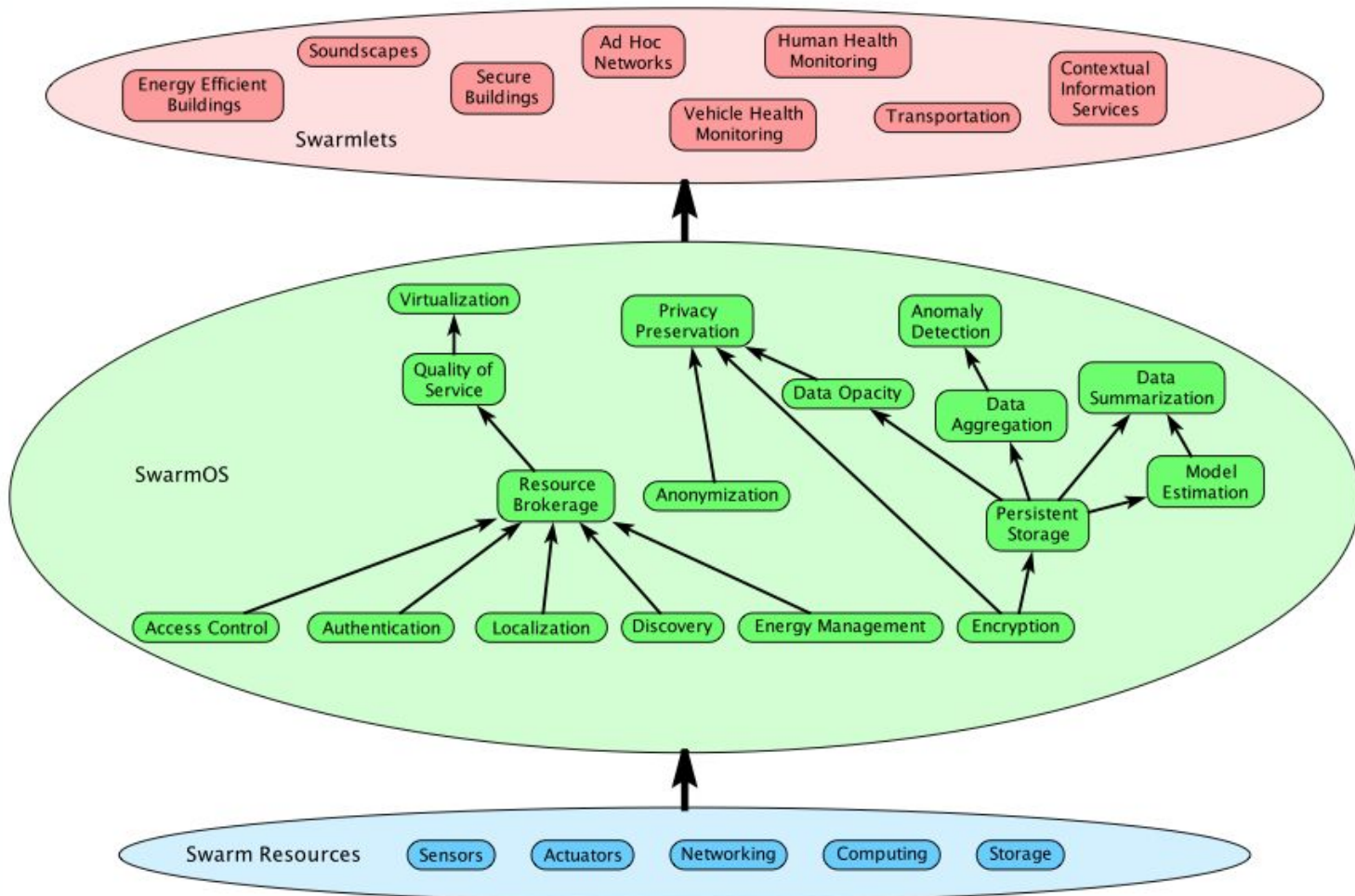


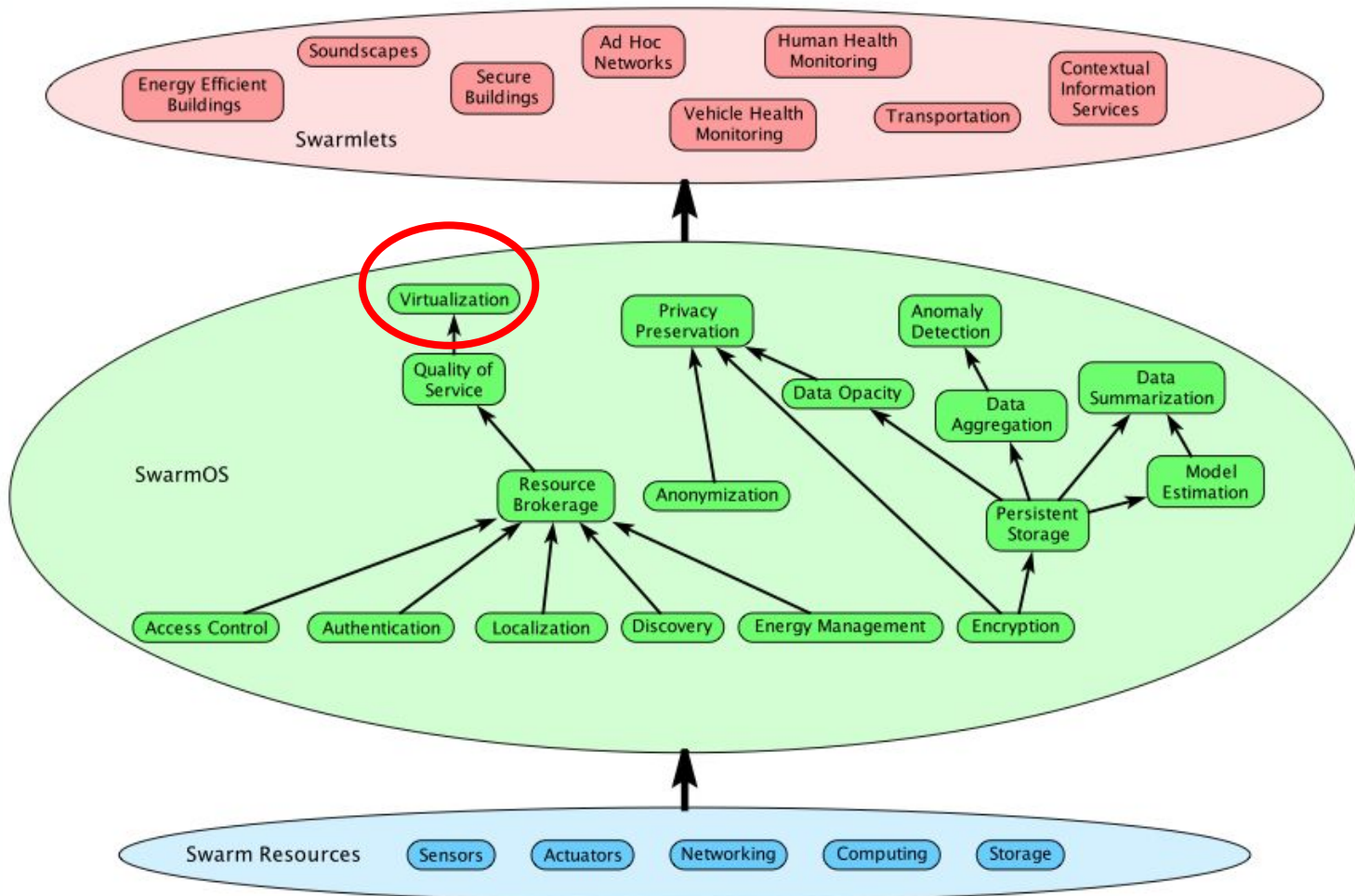
Terraswarm

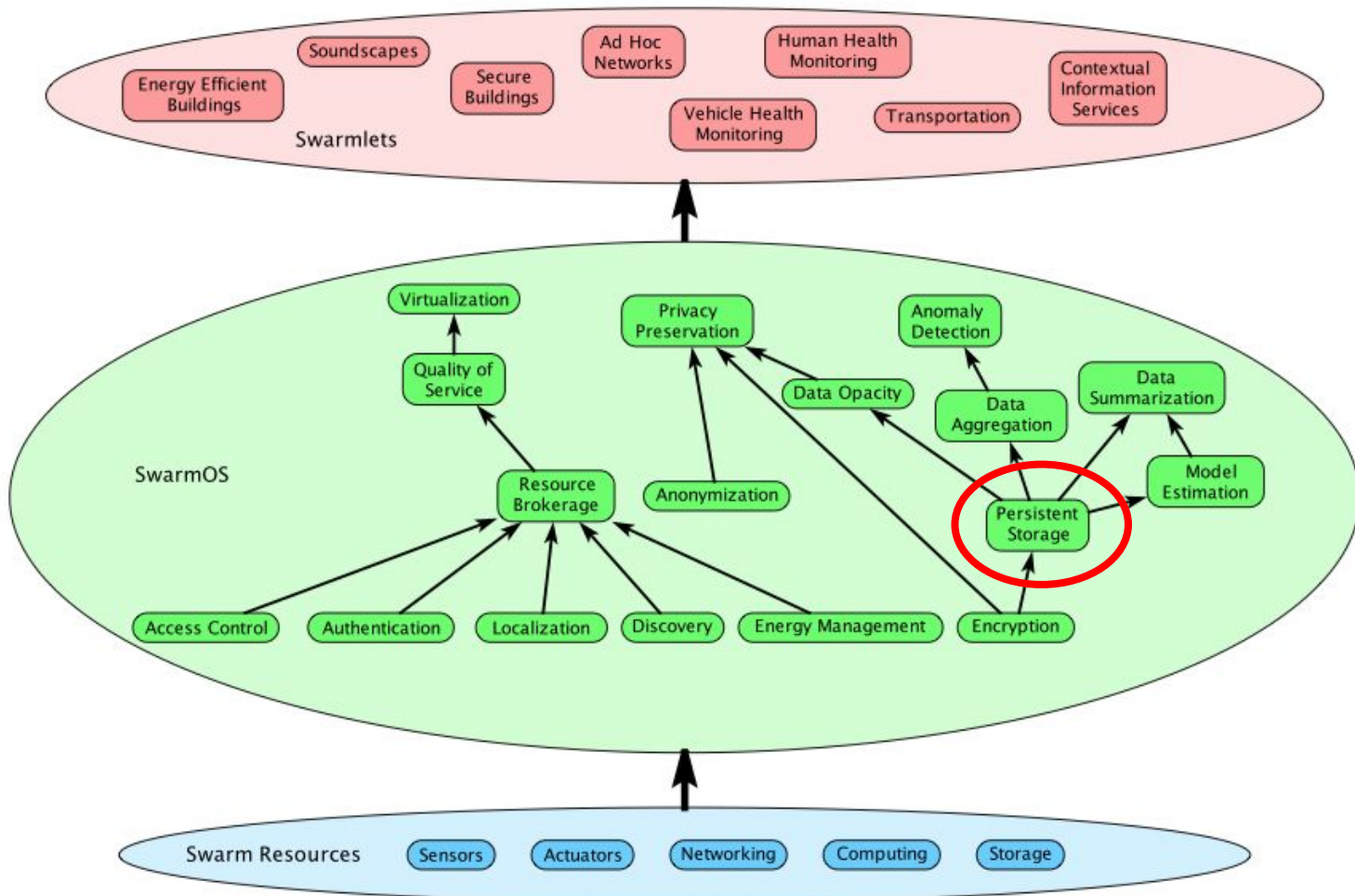
- Terraswarm: encompassing trillions of sensors and actuators deployed across the earth
 - Dynamically recruit resources
 - Aggregate and use that info
 - Dynamically recruit actuation resources
- Deploy readily and rapidly

SwarmOS

- "SwarmOS" highly distributed infrastructure that touches every node in the system
 - Efficiently allocate resources based on complex optimization strategies
 - Allows continual reconfiguration of applications







SwarmBox Hardware

Applications

Swarm Services

SwarmOS Essential Services

Global Data Plane

COTS System Software

SwarmBox Hardware



- Fanless Industrial Computer
- Intel 5th Generation i5 Processor
- IEEE 1588 Ethernet port(s)
- BLE and WiFi
- 8GB DRAM
- 64GB SSD or 1 TB disk drive
- USB

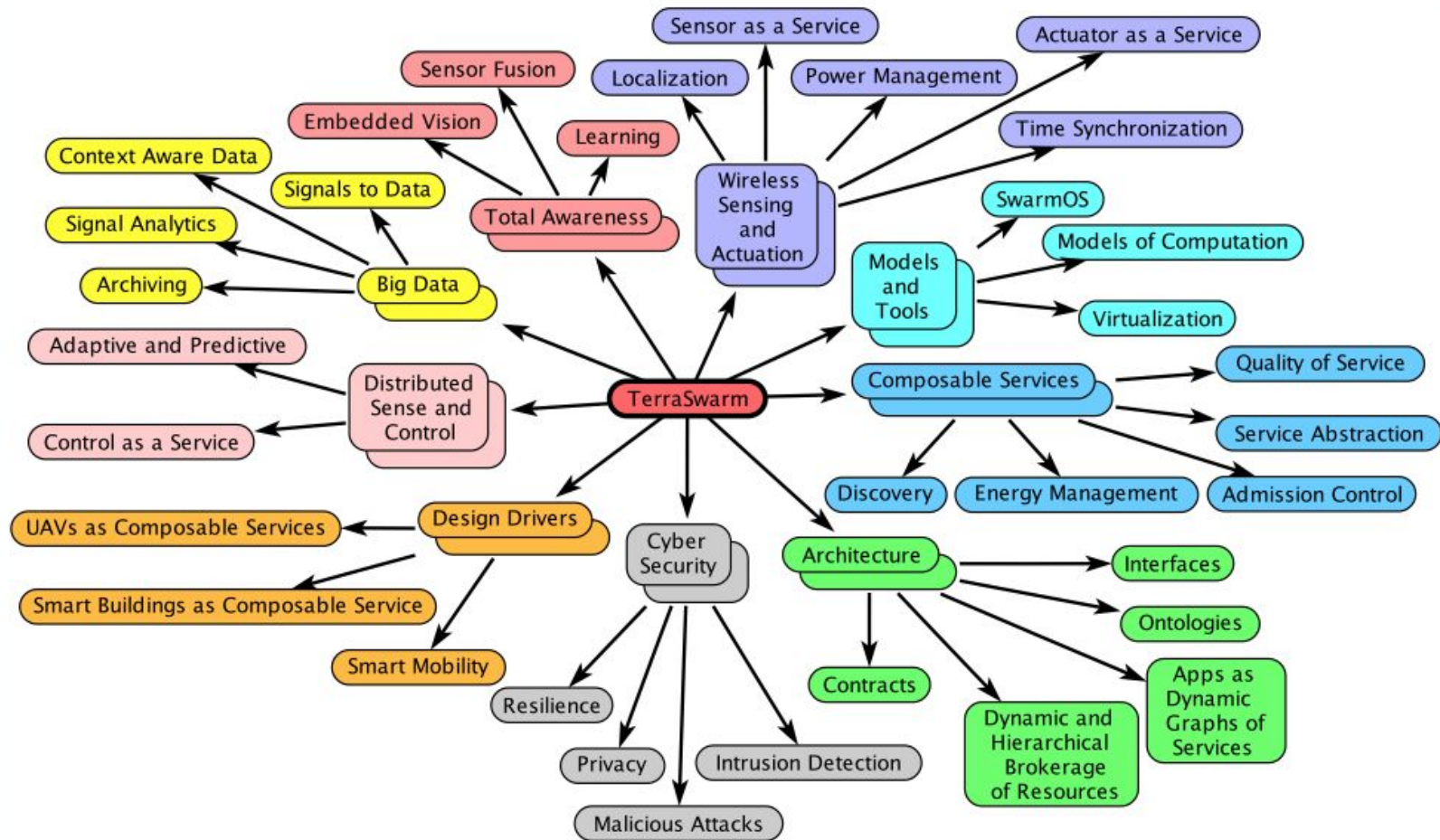
<https://swarmlab.berkeley.edu/research/industrial-cyber-physical-systems/global-data-plane/swarm-box>

Terraswarm Challenges

- Large-scale
- Distributed
- Cyber-physical
- Dynamic
- Adaptive
- Heterogeneous

Solutions

- Data doesn't need to be communicated or stored if it can be predicted from models
 - if its learned, terraswarm can be self sufficient and monitor other devices
- Privacy may be easier to preserve with more data than with less
 - Differential privacy

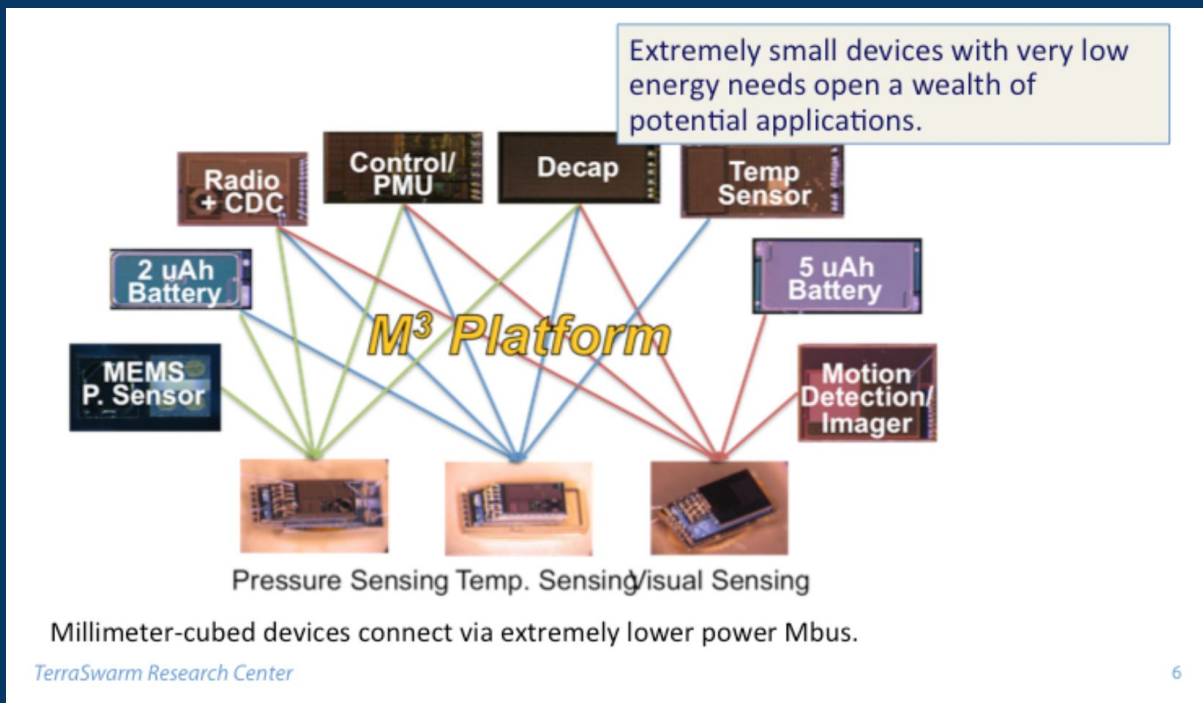


Problem Definition

How can data be efficiently accessed and stored along **without** compromising costs, privacy, & accuracy?

System Design

- design and run time are blurred
- non-interface properties are key
 - i.e node joins or leaves it doesn't interrupt a service



Platform Architecture & Operating Systems

- Service-oriented architecture (SOA)
 - Loose coupling
 - Service abstraction
 - Discoverability
 - Composability
- Virtualization of computing resources is not in conflict with time sensitive and real time services
 - Simulation models will be needed to integrate with deployed systems to adapt in times of stress

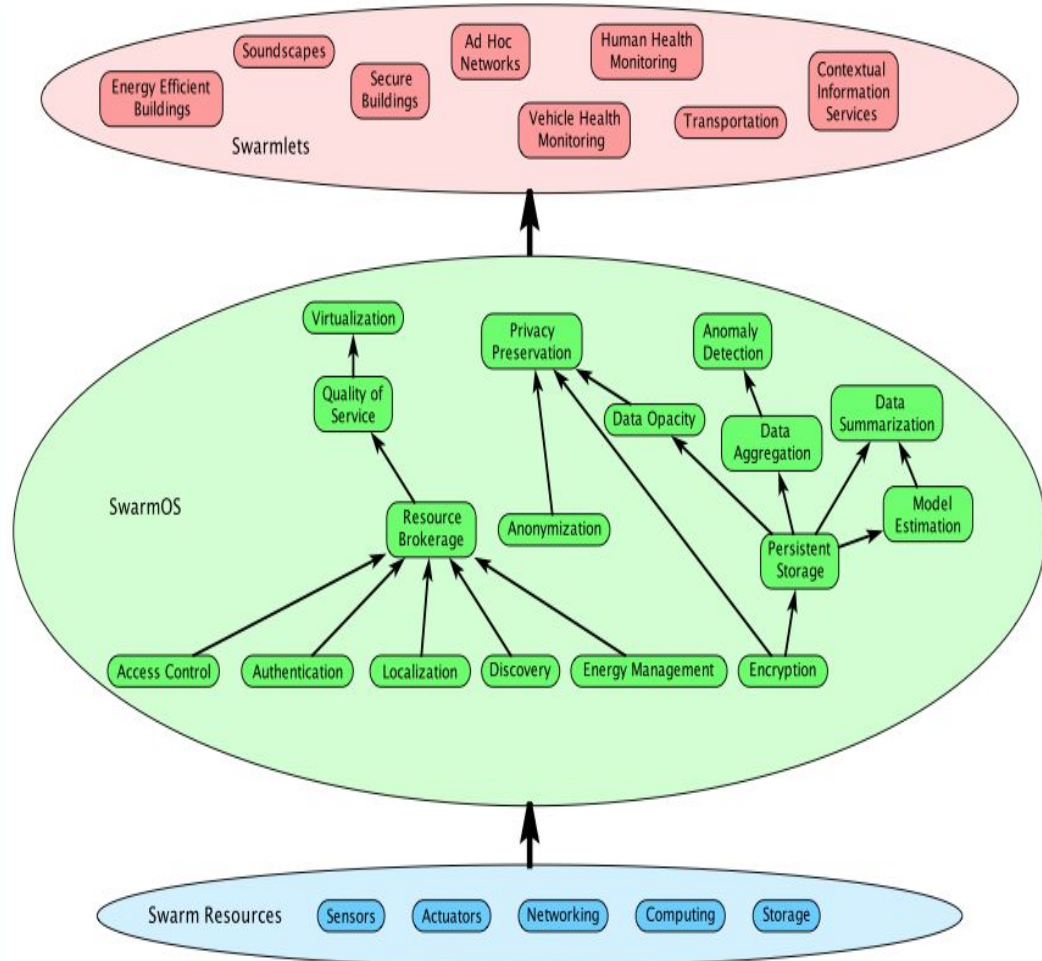
Evaluation: Terraswarm Models

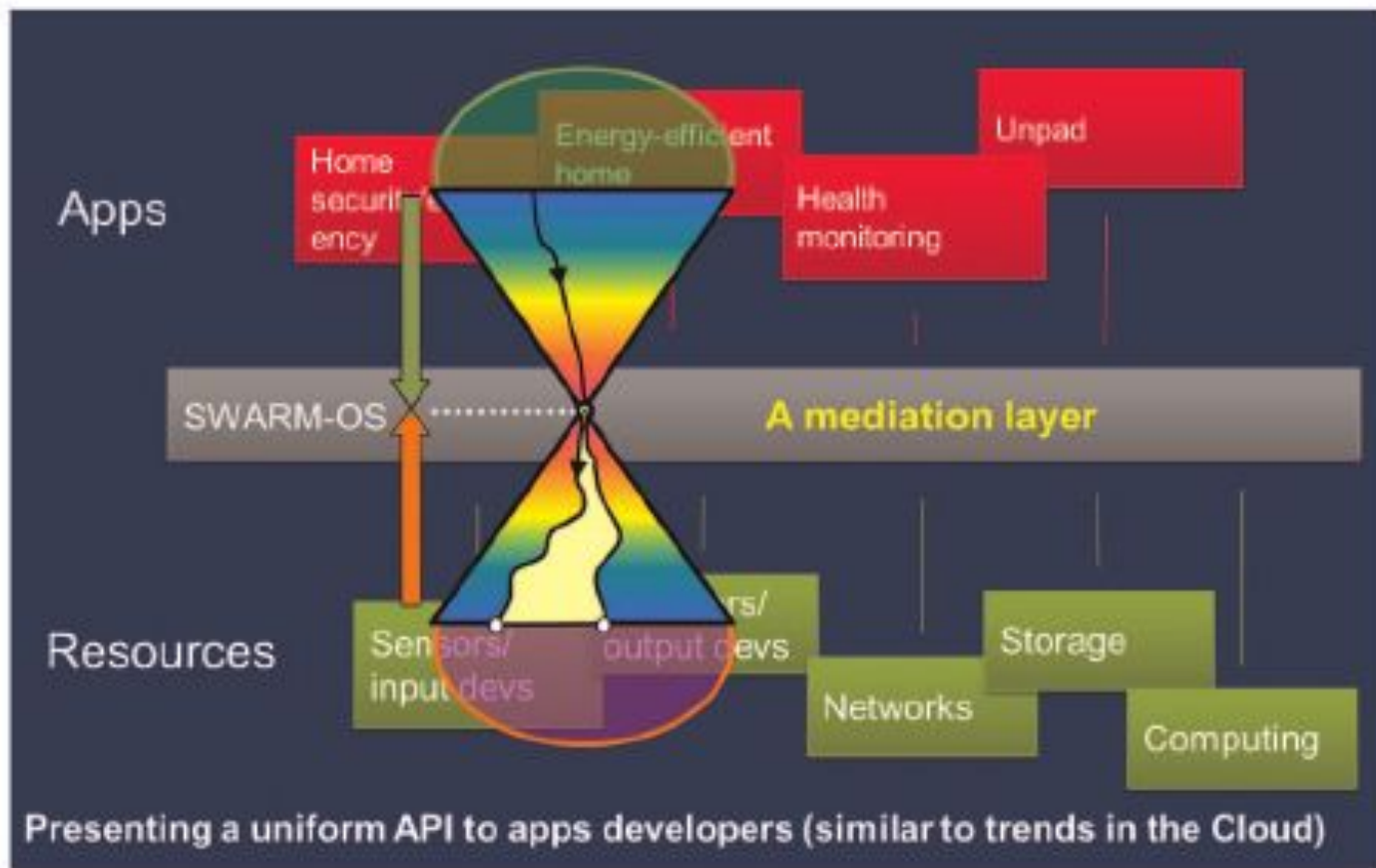
Terraswarm models are designed in hierarchical graphs of components

The edges represent:

1. Communication paths between components
2. Authority relations between components
3. Use relationships (i.e., service x uses service y)
4. Ownership relations
5. Coordination
6. Controllability
7. Observability

1. Communication paths between components
2. Authority relations between components
3. Use relationships
4. Ownership relations
5. Coordination
6. Controllability
7. Observability





Critique

- There should be more info on the specifics of how the devices would function
- It seems up in the air and a lot of hypothesis and assumptions
 - The backbone of the entire paper
- There has been not a lot of recent research to provide more information and progress
- The lack of discussing potential risks is somewhat concerning

Conclusion

- Terraswarm introduces a concept that brings wireless sensor networks to the next level
- Ambitious concept with general high level ideas and thought out components of
 - Privacy
 - Storage
 - User Interactions
- Dynamic change in daily cloud based operations

Discussion

- What is the incentive for IoT device makers to join this?
- If there are two nodes providing conflicting information, how does it choose which one to believe?
- Can we adapt other existing IoT abstractions to work with the swarm concept? For example, how does edge computing fit into all of this?
- How will the swarmOS be able to interface with all the types of embedded devices and cloud providers in order to have a unique platform? Are they expecting everyone to accept and migrate to their solution?