Center for Embedded Networked Sensing

UCLA – UCR – Caltech – USC – CSU

CENS

Scalable Multi-resolution Storage and Search in Sensor Networks

Deepak Ganesan, Ben Greenstein, Denis Perelyubskiy, Deborah Estrin and John Heidemann[1]
Laboratory for Embedded Collaborative Systems, UCLA – http://lecs.cs.ucla.edu
[1]University of Southern California – Information Sciences Institute (USC-ISI)

Can we use existing data handling architectures for wireless sensor networks?

Typical sensor nodes have limited storage capacity

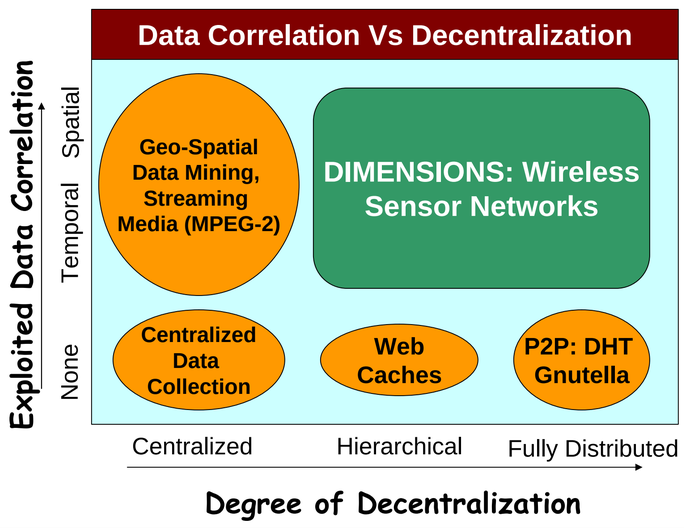
Application	Expected Time to Storage Limit if all raw data were stored	
	Motes	Ipaqs
Habitat Monitoring	Few months	Few years
Building Health Monitoring	Few days	Few months
Contaminant flow	Few months	Few years

Requirements for Scientific Apps

- Long-term deployment, therefore optimized use of energy and storage resources.
- Flexible search for spatio-temporal features.

Existing Techniques are insufficient

- Centralized data collection is too expensive, while in-network storage capacity is limited (left).
- Existing distributed storage and search systems do not exploit spatio-temporal correlations (right).



Need efficient data reduction, multi-scale data processing, and distributed resource sharing

Exploit Scale

Large-scale sensor network deployments offer a significant distributed storage pool, even though individual sensor nodes have limited local storage. Resources should be used effectively to enable long-term deployment.

Approximate Query Processing

Exact query processing is often expensive. In order to be energy efficient, system should provide statistically accurate responses, which can be obtained at lower cost.

Exploit Spatio-Temporal Correlation

In a dense sensor network, significant correlation can be expected and should be exploited to reduce storage and communication requirements.

Gracefully Degrading Query Quality

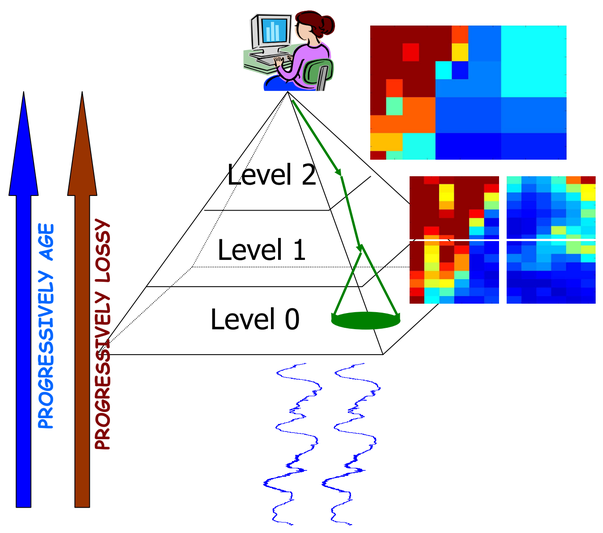
Degrade quality gracefully over time. Accurate responses to queries on recent data can be used to trigger data collection, and time-critical operations. Less accurate responses to queries on older data can be used to understand data trends over time.

Multi-Scale processing

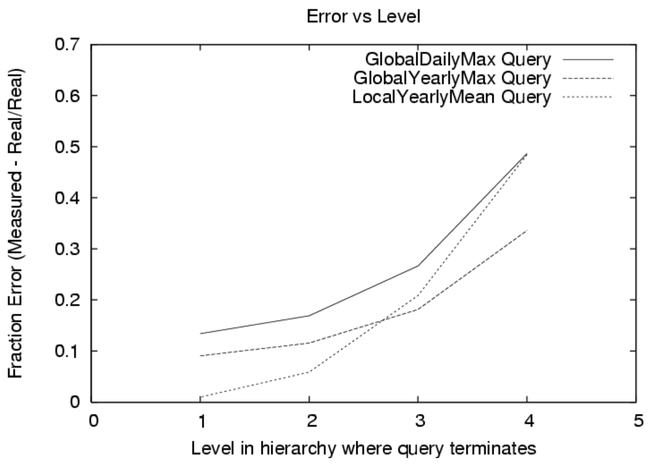
Events exist at certain spatio-temporal scales, for instance, an edge can have a steep or slow gradient. Therefore, the system should support data interpretation over different scales, to extract information from data.

DIMENSIONS: A wavelet-based system for distributed, progressive storage and search

Hierarchical Summarization

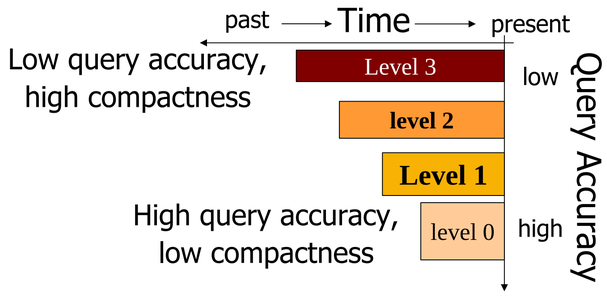


- Construct hierarchy of *lossy wavelet-compressed summaries*.
- Queries "*drill-down*" from root of hierarchy to *focus search* on small portions of the network.

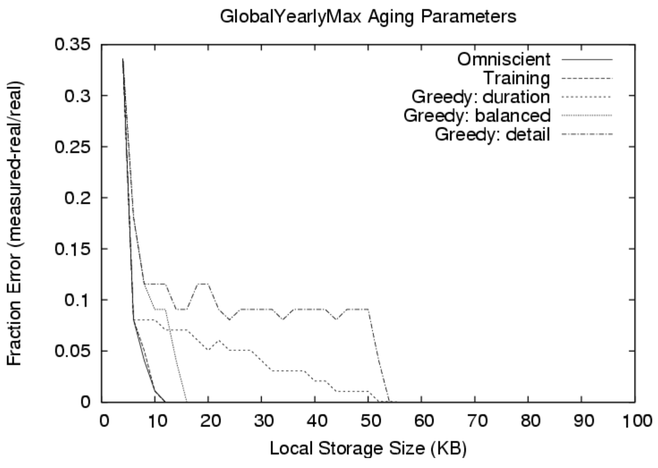


- Excellent query quality is obtained over most queries with sufficient drill-down levels.

Progressive Aging

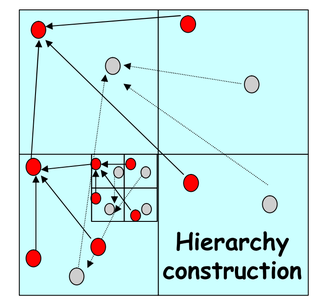


- *Progressively age* summaries along spatio-temporal hierarchy for long-term storage. Retain summaries based on their *utility* for query processing and their *compactness*.
- How do we determine utility of summaries *before* a system is deployed?
  - Use *training* datasets available from prior deployments or initial data collection. Solve constraint-optimization problem to determine aging parameters for summaries.
  - Use *greedy* heuristic that weights summaries depending on expected importance to query processing.

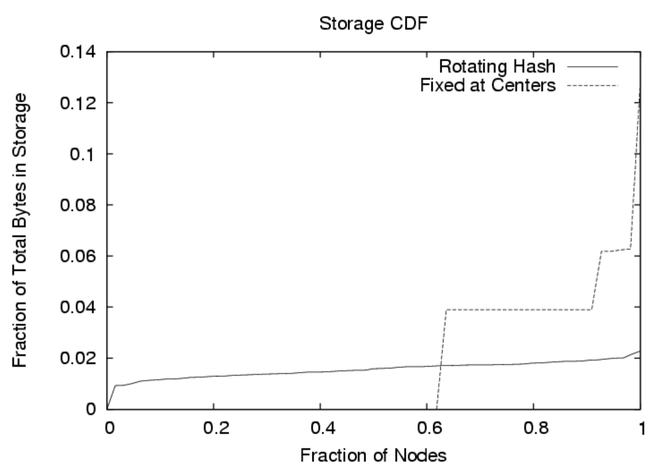


- Training performs within 1% of optimal.
- Greedy performs well for appropriate choice of weight parameter.

Distributed Resource Allocation



- Construct distributed quad-tree.
  - based on previous work: Geographic Hash Tables (Ratnasamy'02), Greedy Perimeter Stateless Routing (Karp'00)
- Balance storage load by periodically picking different cluster-heads at each level of hierarchy.



- Near-uniform storage load-sharing is achieved by periodic cluster-head rotation on uniform topologies.