An Evaluation of Multi-resolution Storage for Sensor Network

SenSys 2003

Deepak Ganesan, Ben Greenstein, Denis Perelyubskiy,
Deborah Estrin, John Heidemann

presented by Roh, John

2004. 5.20

Database Lab. KAIST

Content

- One Line Comment
- Scenario & Problem
- Solution Approach
- Critique
- New Idea

One Line Comment

- **The multi-resolution storage model**
  - providing efficient support for drill-down queries
  - with i) the existing summarization technique and

    ii) a new aging technique of summaries
  - in (communication &) storage-constrained sensor network

Scenario - Precipitation Monitoring

- Let's use sensor network to monitor a rainfall in a certain region…

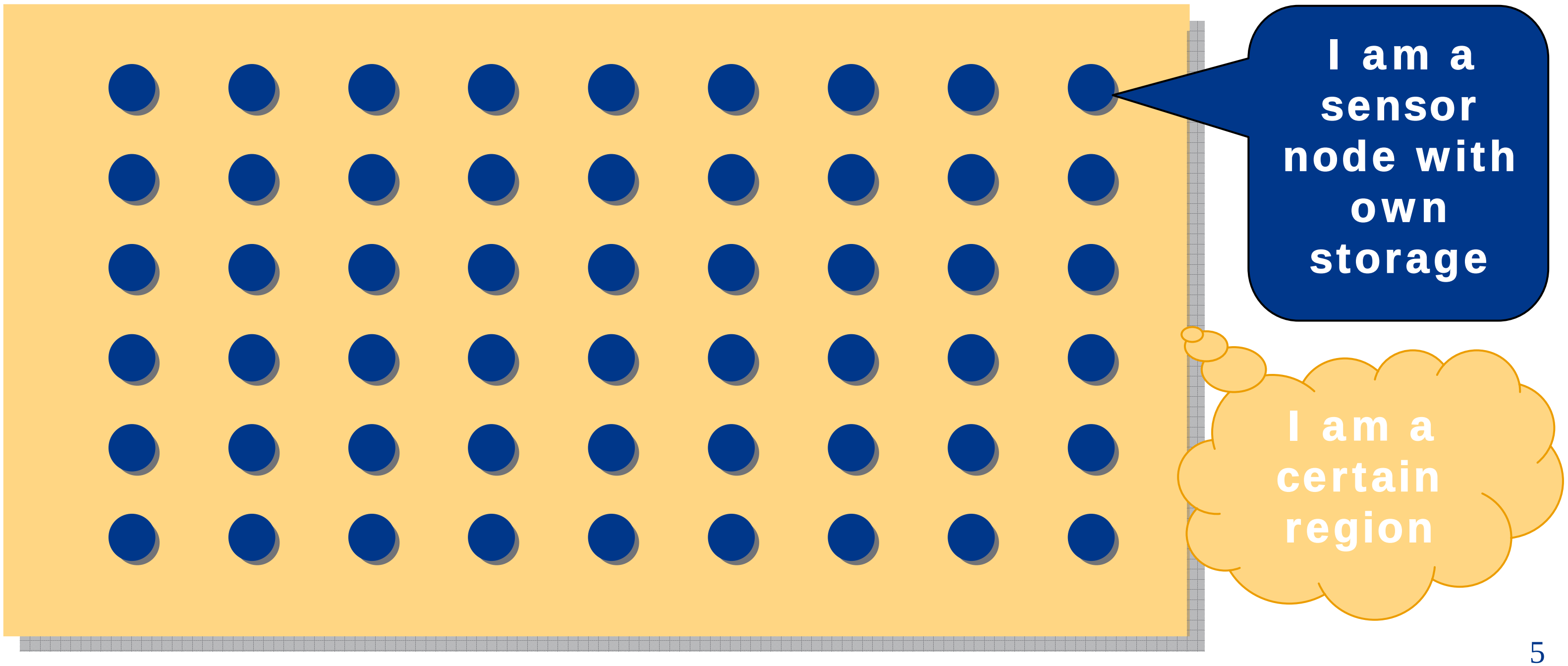
Query Example :

What is the maximum annual precipitation for year X?

Scenario & Problem

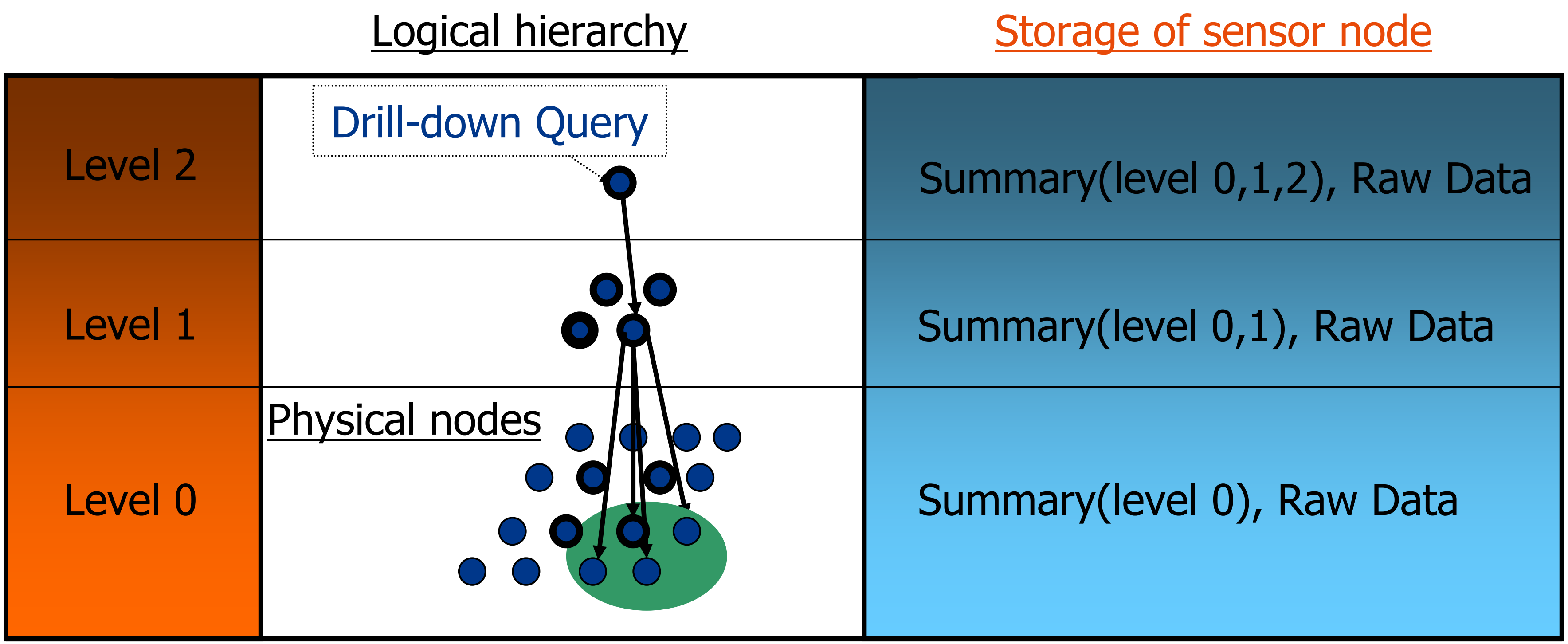
- Let's deploy sensors with own storage(*local-storage* scheme)
- For the monitoring, we may use sensor nodes for a long time
- Then the storage of sensor node may full after some time

➔ "How can we optimize the **storage resource** of each node?"

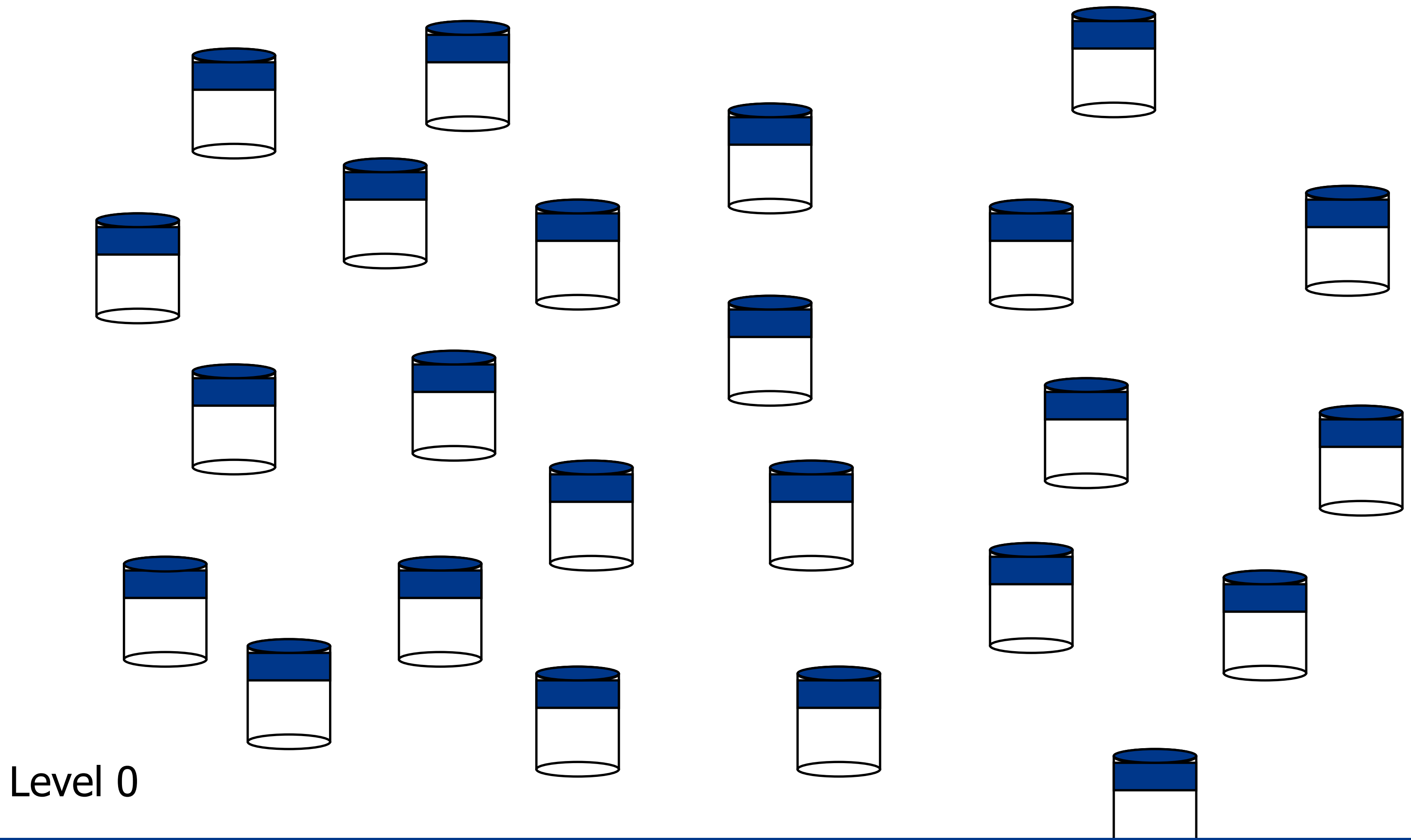


Solution Approach

- Hierarchical(multi-resolutional) storage construction with summarization using a existing technique(wavelet)
- *Aging technique of summaries in each sensor node*



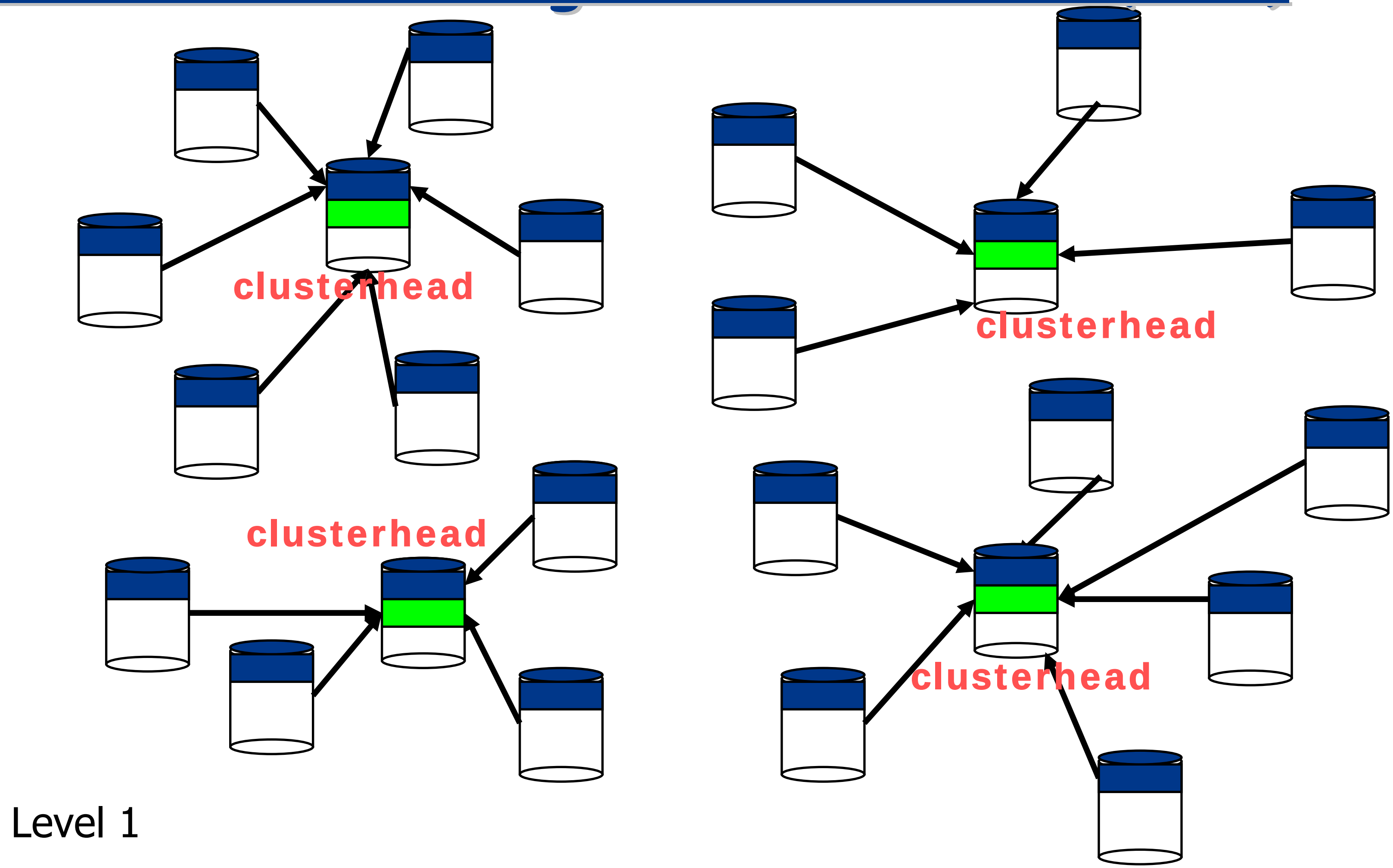
Hierarchical Storage Construction (1/3)



Initially, each node fill up own storage with raw sampled data

Each node constructs summary(level 0) from raw data

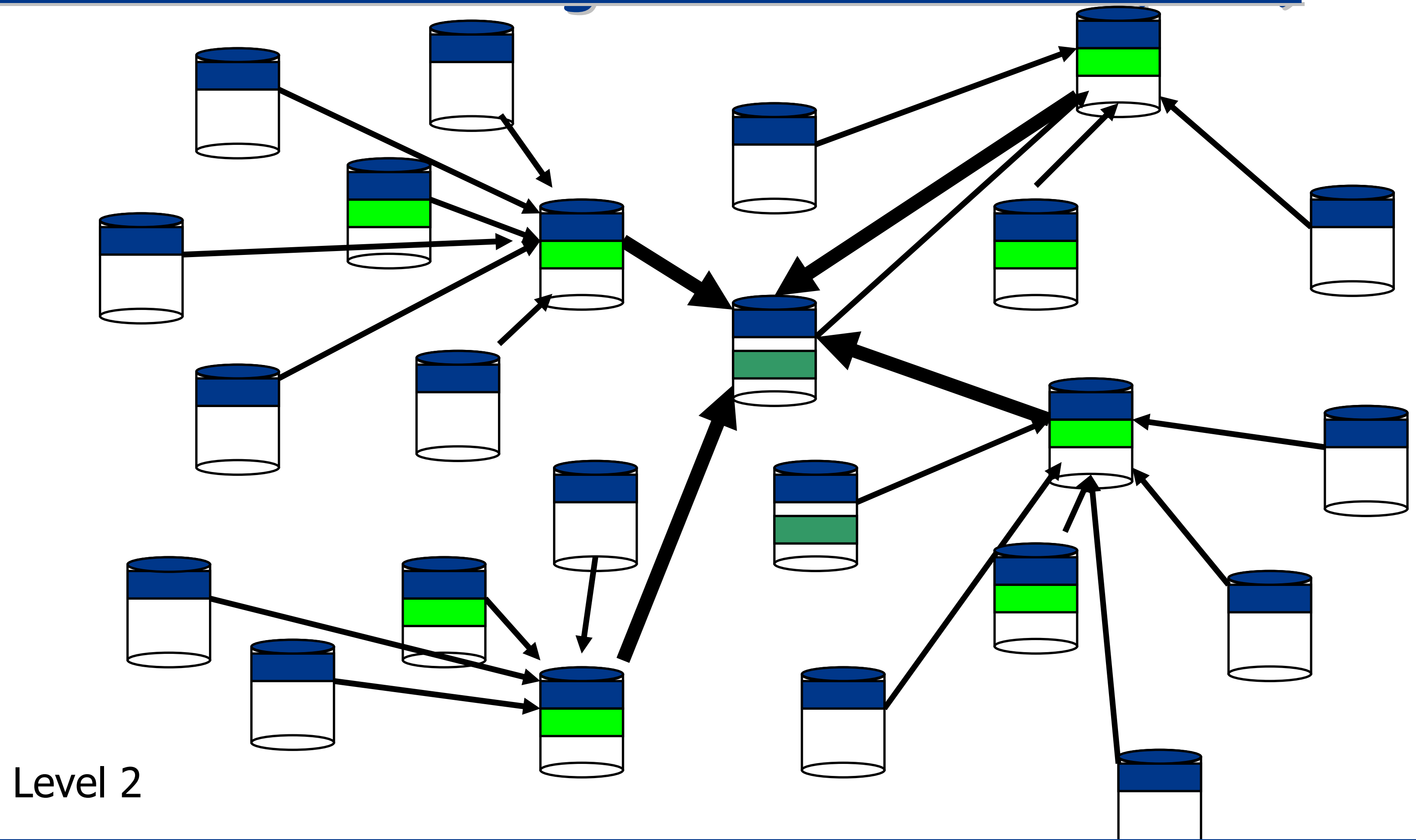
Hierarchical Storage Construction (2/3)



Each node hashes to determine location of clusterhead

Each node sends summary (level 0) to clusterhead

Hierarchical Storage Construction (3/3)



(Each node hashes to different locations over time to distribute load among nodes)

Each clusterhead sends summary(level 1) to upper clusterhead so that upper clusterhead constructs coarser summary(level 2)

Storage Optimization (1/4)

- "With a limited storage, how can we optimize the storage in sensor node?"
  - That is, how we allocate storage at each node to summaries at different level?

Total storage of clusterhead (level 2)

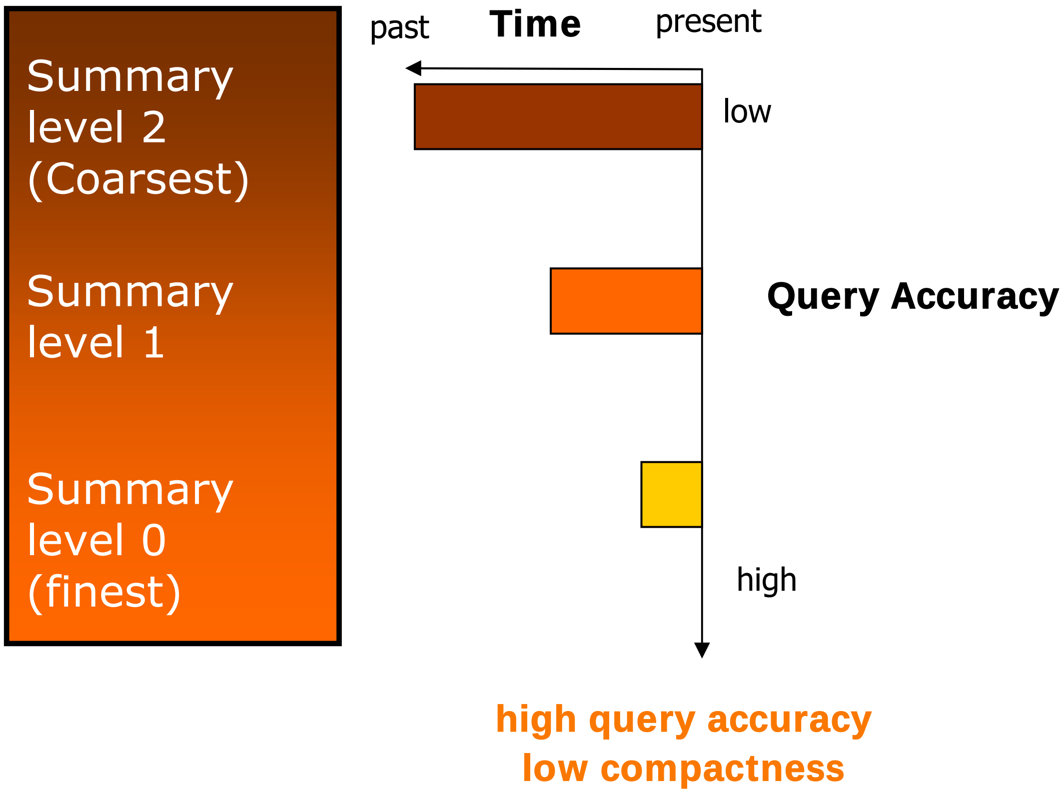
Summary level 0(finest)	Summary level 1	Summary level 2(Coarsest)

5 : 2 : 3 is good?

1 : 3 : 6 is good?

Storage Optimization (2/4)

- Graceful Query Degradation
  - More accurate responses to queries on recent data
  - Less accurate responses to queries on older data

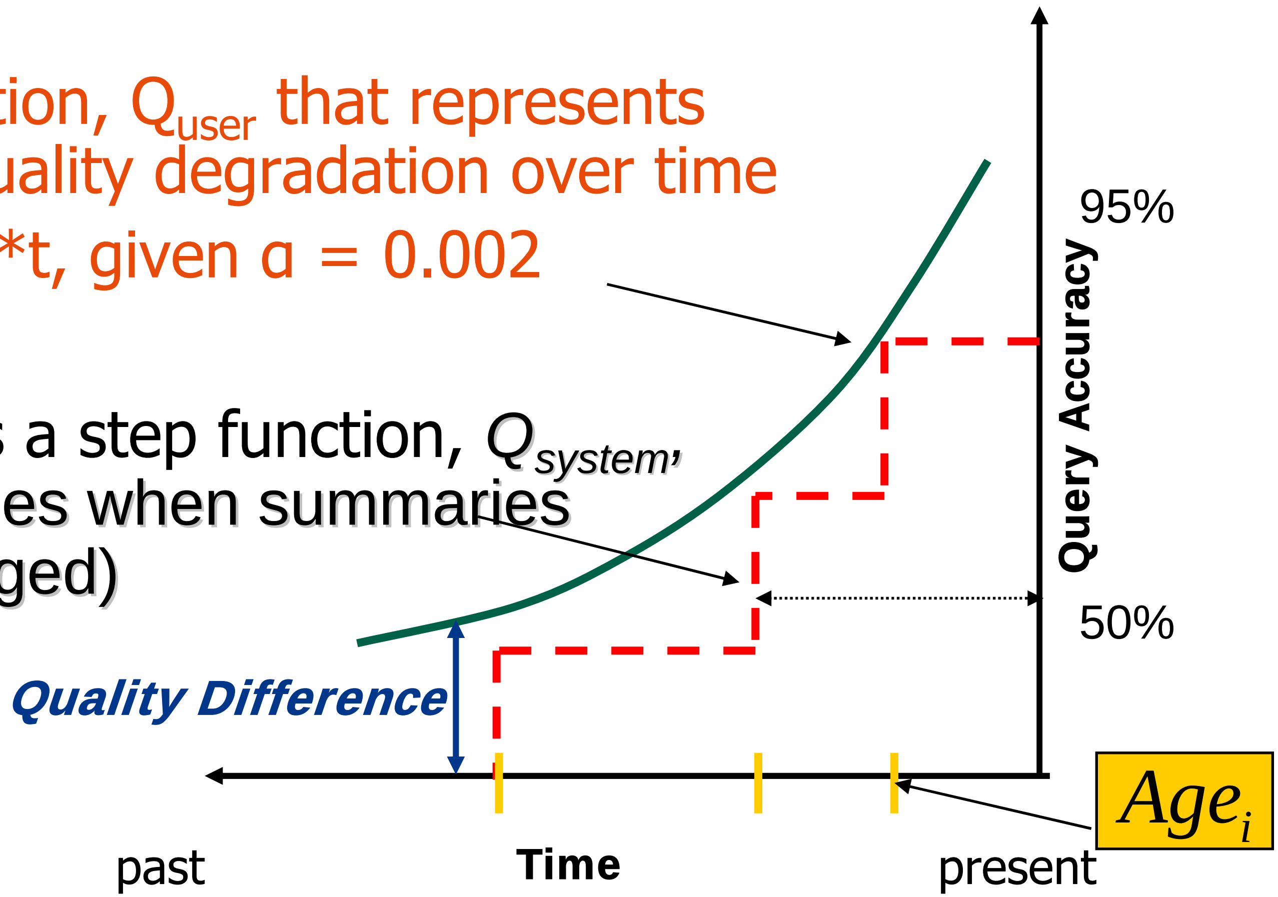


Storage Optimization(3/4) – Aging Summary

Let’s use a function, Q_{user} that represents desired query quality degradation over time

e.g. $f(t) = 1 - \alpha * t$, given $\alpha = 0.002$

System provides a step function, Q_{system}, with steps at times when summaries are destroyed(aged)



Objective: *Minimize worst case difference between user-desired query quality (blue curve) and query quality that the system can provide (red step function)*

Storage Optimization(4/4) – Aging Summary

Full a priori information

Omniscient Strategy
Lower bound for query error Use
all data to decide optimal allocation

Training Strategy
can be used when small training
dataset from initial deployment

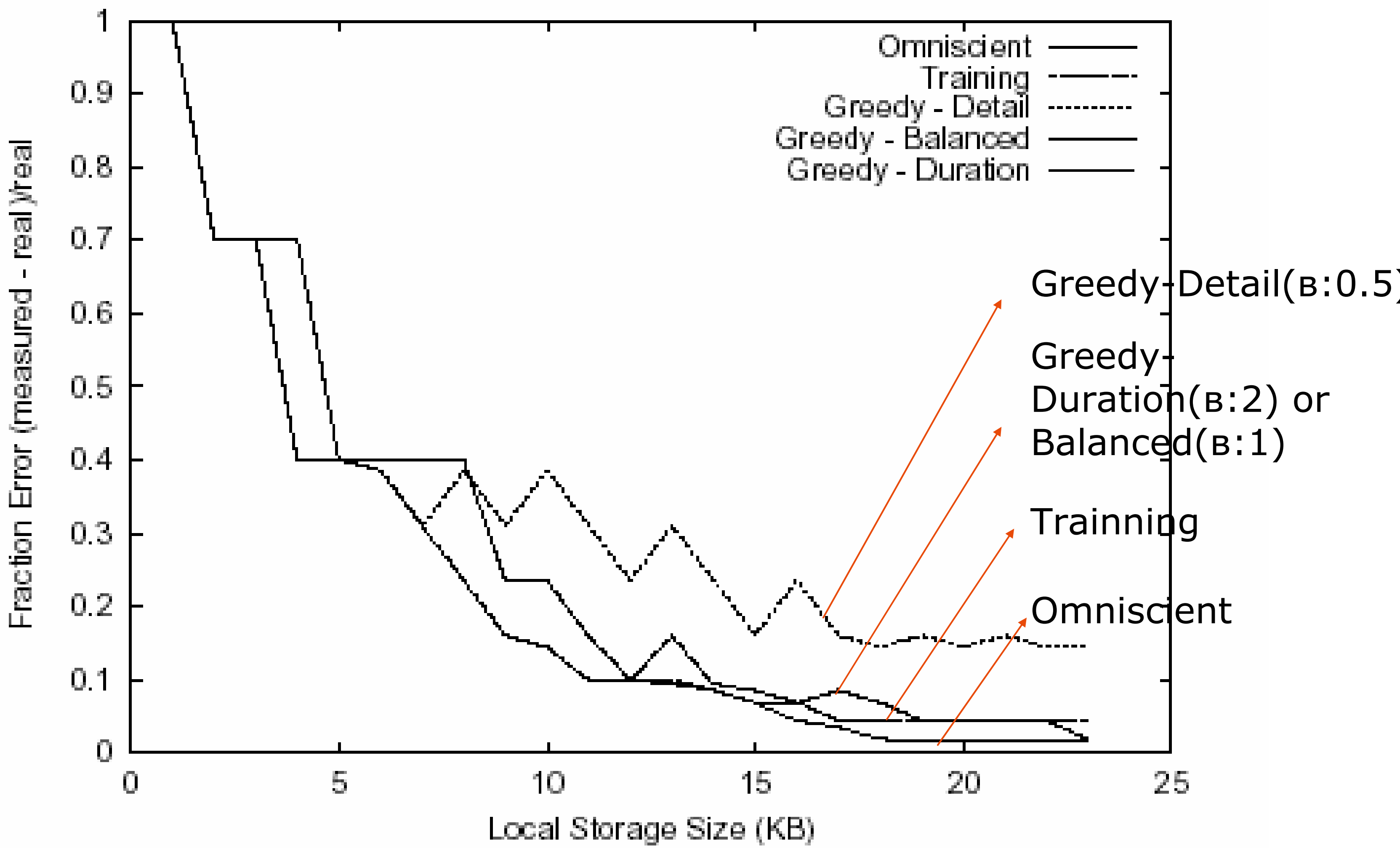
Greedy Strategy
when no data is available, use a simple
weighted allocation to summaries
e.g.
B:1 ➔ finest:finer:coarsest = 1:2:4
B:2 ➔ finest:finer:coarsest = 4:2:1

No a priori information

Assumptions

- Grid deployment of sensor node
- Homogeneous sensor node
- *Local-storage* scheme : Each sensor with own storage
- Continuous measurements rather than events
- A lot of data generated by sensor node over time

Simulate Result



Simulate Result – Setting

- Data set : geo-spatial precipitation dataset
- Number of sensor node : 15 X 12 = 180
- Period of summary creation : 3 epochs(years)
- Raw data bytes : 3 X 365 X 2 =2190 bytes
- Compression rate (c0 : c1: c2: c3 = 6: 12: 24: 48)
- Q_{user} i.e $f(t) = 1 - a*t$, given $a = 0.002$
- Query : What is the maximum annual precipitation for year X?

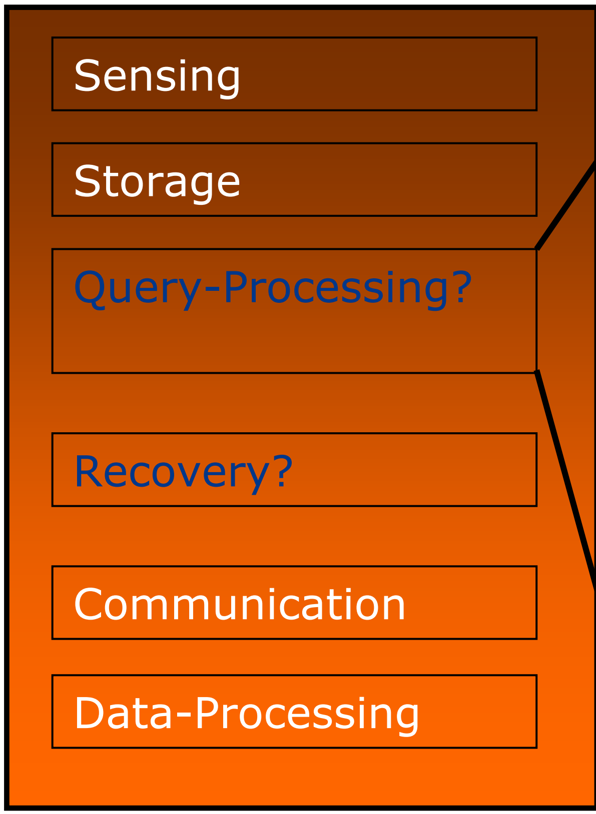
Critique

- Strong Point
  - New consideration of sensor network as communication & **storage-constrained** environment
- Weak Point
  - **Difficulty in setting parameters**
    - There are too many parameters to apply the proposed model to real sensor network that have an influence on performance
      - Period of summarization
      - Compression ratio for each resolution
      - User-specific aging function
      - Resolution-bias(B in greedy strategy)
  - **Experiment Size**
    - The author argues "the proposed model is suitable for a large data application" BUT the experiment are performed with somewhat small dataset
    - Node number used in experiment are too small (180 sensor node)

    ➔ Experiment of a larger scale may be need to identify the usefulness of proposed method

New Idea (1/2)

- Sensor Node



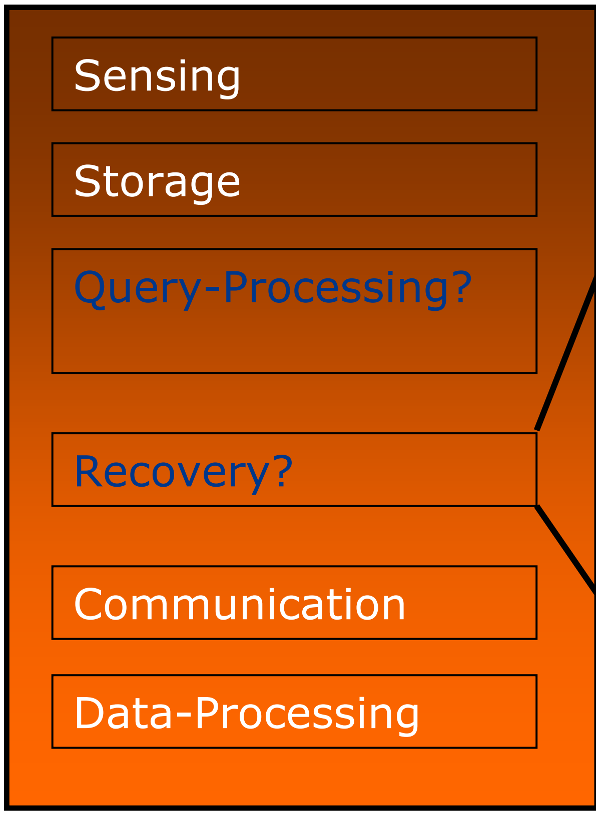
To process large number of event-based queries, after sensor node sensing, sensor node may need to check if the sensed data satisfies any of large number of events.

If we build some structure such as "index of events" i.e. index of trigger, then above situation will get better?

How about we design that considering the time which is not appeared in traditional DB triggers.

New Idea (2/2)

- Sensor Node



After sensor node failed and re-powered, sensor node may not know what to do

only if there is no structure on event-based queries.

By building and maintaining the “log-based” index of events on local disk in sensor node, above situation will get better?