Project: Sensor Network Database

* Description

In this project, we build a remote environment monitoring application using a network of embedded sensors. The physical environment under monitoring (e.g., InfoLab) is modeled as a relational database with both static data from the environment entities (e.g., PCs, appliances, GRAs(!), ...) and dynamic data from the sensors. The entire database is maintained within the sensor network. Remote clients connect to the "sensor network database (DBSN)" via a number of base stations, and issue embedded SQL queries as well as GUI-based SQL queries. Queries are injected into DBSN, queries are resolved by in-network query processing, and query result is propagated back to the remote client through the corresponding base station.

The project people will be organized into 4 groups, each to implement one sub-system of the monitoring application:

(A) Range Query Processor (3 students)
(B) Aggregate Query Processor (3 students)
(C) Join Query Processor (3 students)
(D) Base Station Server (2 students)

Groups A-C implement the corresponding in-network distributed query processing application. Each group should implement at least two approaches and compare their performance. Group D provides the server at the base station, which is the interface between remote clients and DBSN.

Project is implemented in two phases (see the time schedule). At Phase (I), the query processors are implemented and tested via the TinyOS simulator (TOSSIM). Detailed empirical comparison of the query processing approaches are due at the end of this phase. Also, at this phase Group D implements and verifies the base station server that provides a common interface for the query processors at the back-end and a well-defined API at the front-end. At Phase (II), the processors are integrated with the server and the application is tested within the actual sensor network embedded in the environment.

* Administration:

- Each group has a group leader to coordinate with the project leaders.
- Grading policy:
  > 60% Phase I, 40% Phase II
  > Contributions of each student is evaluated independently at the

project team level, group level, and individual level.
  > Group contributions are evaluated based on the performance measures (code size, energy efficiency, response time), originality, and completeness.
- Template test cases for the DBSN schema, instance relations, and instance queries are provided (see test cases).

* Resources and Related Links

- TinyOS tutorial:
http://webs.cs.berkeley.edu/tos/tinyos-1.x/doc/tutorial/index.html
- TinyOS Boot Camp:
http://webs.cs.berkeley.edu/tos/presentations/Boot_Camp/boot1.ppt
http://webs.cs.berkeley.edu/tos/presentations/Boot_Camp/boot2.ppt
http://webs.cs.berkeley.edu/tos/presentations/Boot_Camp/boot3.ppt
http://webs.cs.berkeley.edu/tos/presentations/Boot_Camp/boot4.ppt
http://webs.cs.berkeley.edu/tos/presentations/Boot_Camp/boot5.ppt
- Sensor network platform:
http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/6020-0042-01_A_MICA2.pdf
http://www.cs.berkeley.edu/~jhill/papers/MICA_ARCH.pdf
- TinyDB:
http://telegraph.cs.berkeley.edu/tinydb/
- Habitat Monitoring sample application:
http://www.greatduckisland.net/index.php

* Time Schedule

- 9/2: Students should make their group and decide which sub-system they want to implement. The group leader should send the name of the sub-system and the name of group members to project leaders before midnight.

- 9/18: Each group presents their sub-system proposal including a survey about the research work on their part and the approaches they want to develop.

- 10/16: Each group presents a project report describing one of implemented approaches.

- 11/6: Phase I ends. The simulation part should be presented. The interfaces will be tested.

- 12/4: Phase II ends. The whole system is presented through an extensive demo in infolab.

* Test Cases:

Some sample database schemas and different queries will be provided to the groups as test cases.