Objectives:

The objective of this part of the effort is to provide a shared repository, called Senstore, for sensor and structural data. Our aim is to design Senstore so that it

1. Supports high throughput for storing sensor data
2. Permits temporal queries for sensor data. It should be possible to query for historical sensor data, even if the set of sensors or their configuration changes over time
3. Provide access control features since structural data about bridges can be sensitive
4. Link sensors with structural data
5. Integrate with external data sources, such as weather data

Technical Progress: (1 page)

We started out with the premise that to the extent possible, we will attempt to use a standard database and a standard interface, e.g., SQL, for Senstore. We picked Postgres since it is open-source and is a robust database system. One concern with Postgres was whether it would be adequate in terms of throughput for receiving sensor data at high data rates. Also, we wanted to be able to integrate external data sources down the road into the repository.

Our first finding was that Postgres does work with external data sources. Essentially, we were able to create virtual tables inside the database. When these tables are used, external sources are accessed to populate the tables as needed.

We then created a schema to represent sensors and their data. To handle temporal queries, the schema allows time to be associated with sensor and transducer configurations. We can thus ask a query about the sets of transducers, and their attributes, at any given point of time. Both sensors and transducers can be reconfigured over time and historical queries can be supported.

A challenge we faced is how to identify the location of a transducer. One possibility was to use (x, y, z) coordinates. But, in practice, that is difficult to do. Coordinates of a sensor are difficult to determine precisely. Our proposed solution is to associate a transducer as being relative to a particular *node* in the bridge geometry. A *node* is a technical term here representing a point of interest in the bridge geometry. For example, a beam may be defined by connecting two nodes. We plan to discuss this design further with structural engineers in our group.

The above solution required bridge geometry to be also represented in the database. We took a Microsoft Access database that was provided by SC Solutions and adopted it to Postgres use. The overall schema is posted to CTools for discussion.

Discussions about Senstore with SC Solutions led us to considering alternatives to Postgres for the sensor data itself. The problem is that sensor data can be very bursty. One can get samples generated at 200 Hz from each transducer. Attempting to import this volume of data may push a database system to its limits, especially on commodity hardware. After evaluating the performance of Postgres, we decided to adapt a mixed storage model as our first strategy that we will try out. It consists of a system called HDF5 from University of Illinois for storing sensor data, and Postgres for all the meta-data and bridge geometry. HDF5 was found to be approximately 1-2 orders of magnitude faster in throughput than Postgres. A downside of HDF5 is that it is not as robust against computer crashes and does not support a powerful query language like SQL. So, this strategy will require further validation.

Most significant Technical Advancement to Date

The most significant technical advancement is to create a schema that integrates bridge geometry with sensor-related meta-data. For the first time, it becomes feasible to identify the locations of sensors with respect to bridge geometry and use a simple SQL front-end for queries on both sensor data and bridge geometry.

Impact: (no more than ½ page. All presentations/papers)

The work is leading to interactions between computer scientists and civil engineers on data representation and abstraction issues. Atul Prakash from Computer Science at the University of Michigan visited SC Solutions in California and met Gwen van Der Linden and Abbas Emami-Naeini to discuss the database design. As a result of that discussion, the two teams exchanged technical data and converged on a strategy for database design. Two project reports are expected to result from that collaboration and are currently in progress. The first report, led by SC Solutions, will detail the overall architecture of the system and the planned software strategy. The second report, led by University of Michigan, will define the database schema for the system and the design rationale behind it. The initial version database schema has already been posted to CTools, the shared workspace between the teams. A presentation on the database system, called Senstore, was made at the team meeting.