Demo Abstract: Sensor Fusion and Correlation

Todd Sproull, Richard Hough, John Lockwood
Dept. of Computer Science and Engineering
Washington University in Saint Louis
Saint Louis, MO, 63105, USA
{todd, rh3, lockwood}@arl.wustl.edu

Christopher Zuver, Kent English, John Meier
Boeing Corporation
Saint Louis, MO, 63166, USA
{christopher.k.zuver, kent.l.english, john.l.meier}@boeing.com

Categories and Subject Descriptors: C.2.1 [**Network Architecture and Design**]: Distributed networks; C.3 [**Special-Purpose and Application-Based Systems**]: Real-time and embedded systems

General Terms: Experimentation, Design

Keywords: Sensor Fusion, Data Correlation

The sensor fusion architecture is a General-purpose Aggregation Processor (GAP) designed to bridge the gap between low-level sensor data and the high-level knowledge needed by the backbone. By examining both the incoming data from attached sensor nodes and the interests of users on the network, the model reduces overall transmission costs by keeping local event information at the source, only reporting the higher-level alerts and knowledge to interested parties.

The sensor fusion architecture possesses three distinct advantages worth noting.

- Sensor fusion allows for the network to report events between several different types of sensors using a common communication API. This architecture uses XML messages to report sensor readings.
- The architecture allows for a correlation between events in a specified time window. By combining base predicates together into a complicated event sentence, a user can receive a more accurate description of the situation at hand.
- This system reduces the amount of overall communication traffic when deployed in a sensor network by creating a local aggregation filter between the individual sensors and the end user.

In this demo we will deploy the sensor fusion model in a wireless sensor network designed to model a real-world application. The network will consist of multiple technologically distinct sensors including Micaz motes, RFID sensors, and an intelligent web camera. Each of these pieces will inject XML traffic into the network, announcing various status and event information that the GAP will then parse and filter as appropriate. The motes will be connected to the network through a Crossbow Stargate running TOSBase. The intelligent web camera and RFID door reader will communicate to the server over a 100MBit Ethernet connection. The server itself will run on an embedded Linux computer inside of a GlobalVelocity 1000 intelligent gateway device.

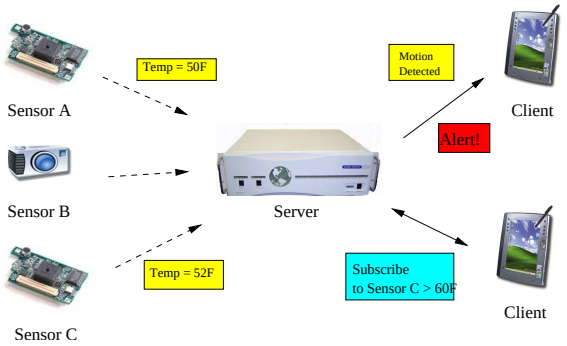

Figure 1: High level view of sensors communicating with clients and the server

For the demo the server, 3 MicaZ motes, an RFID reader, and one laptop will be placed onto the provided 6' X 2' table. Each of these in turn will be connected to a single shared 100MBit Linksys 5-port switch to form the demonstration network.

To represent a standard use case, the demo team will communicate to the GAP through a remote Tablet PC running a Java GUI client. Utilizing this interface, the client will register for four separate event services advertised as available by the sensor fusion server. Alert 1 notifies the client when an RFID match occurs on a specific RFID tag. Alert 2 reports to the client all events from the web camera. Alert 3 indicates a temperature reading exceeding a given threshold. Alert 4 is the correlation between alerts 1 and 2 occurring within 5 seconds of each other. To demonstrate accurate filtering by the server in accordance with the client's needs, we will perturb each of the sensors to generate a stream of events across the network.

Acknowledgment

This research has been supported by the Boeing Corporation.

SenSys'05, November 2–4, 2005, San Diego, California, USA.
ACM 1-59593-054-X/05/0011.