The CarTel Mobile Sensor Computing System

V. Bychkovsky, K. Chen, M. Goraczko, H. Hu, B. Hull, A. Miu,
E. Shih, Y. Zhang, H. Balakrishnan, and S. Madden
MIT Computer Science and Artificial Intelligence Laboratory
The Stata Center, 32 Vassar St., Cambridge, MA 02139
http://cartel.csail.mit.edu

Categories and Subject Descriptors

C.2.4 [**Computer-Communication Networks**]: Distributed applications

General Terms

Design, Measurement, Algorithms, Experimentation

Keywords

Networks, mobility, connectivity, delay tolerant databases, query processing, data visualization, network management

1 Introduction

Worldwide, there are over 600 million automobiles on the road. Each automobile is a potentially rich source of sensor data, with the current generation of cars having over 100 sensors. Unlike many other mobile platforms, an automobile is a resource-rich environment that can support relatively robust computation and communication systems. More importantly, because automobiles interface with a vast amount of the physical world and are well-integrated into our daily lives, they are uniquely positioned to enable a broad range of sensing applications.

What can we do with 600 million mobile computing units (cars), each with tens of sensors, and on which we can place large amounts of computation? Here are some classes of applications that would arise if we expanded the reach of today's Internet-based computing substrate to include automobiles. In all these applications, cars are information sources.

1. *Traffic monitoring and route planning.* Suppose we tracked the location of every car (suitably anonymized) once per second using GPS. We can use this information to develop statistical models of traffic delays at various times of day on different road segments. Suppose you want to leave your home for the airport to catch a flight at 8:00 am. Which of the four different routes to the airport should you take?

2. *Preventive maintenance and diagnostics of cars.* By tapping into the on-board sensors using the standard CAN (controller area network) interface, and by attaching a variety of external sensors, we can monitor and report internal performance characteristics such as emissions, gas mileage, tire pressure, suspension health, etc. These reports can be combined with historical data, highlighting long-term changes in a car's internals, and correlating a given car's information with other cars of the same vintage to detect anomalies in a car's performance.

3. *Civil infrastructure monitoring.* When equipped with additional sensors to sample vibration and other conditions, cars can act as excellent "probes" to sense road conditions. Assessing and reporting road surface conditions such as potholes, oil spills, flooding, and ice can help cities and towns identify roads that need repair at relatively low cost, and help drivers to learn about hazardous driving conditions.

To enable these types of applications and others, we propose a reusable data management system, called *CarTel*, for querying and collecting data from intermittently connected devices. Our platform provides a dynamic query system that allows for both continuous and snapshot geo-spatial queries over car position, speed, and sensory data as well as both a low-cost and high-bandwidth substrate for communicating with a large network of mobile devices. What follows is a more detailed treatment of the key components of the CarTel mobile sensor computing system.

2 System Overview

CarTel relies on three key underlying technologies:

Figure 1 illustrates the basic architecture of the CarTel system. Each car is equipped with a *CarTel node*, which is an embedded PC outfitted with a variety of sensors and software for data collection. As cars drive around, they collect data, such as the car's current GPS location, WiFi availability, and engine performance anomalies. Sensor data is uploaded and control messages are downloaded using intermittent wireless connections (e.g., 802.11 hotspots, Bluetooth cellphones, or other CarTel-enabled cars). This information eventually reaches the Internet, where it is delivered to our data management interface, the AutoPortal, for visualization, analysis, and browsing.

1. *AutoPortal:* Server software that provides data management, visualization, and web-based querying. This software requests data from remote nodes, aggregates reports arriving from those nodes into a coherent picture of current conditions, and visualizes that data.

SenSys'06, November 1–3, 2006, Boulder, Colorado, USA.

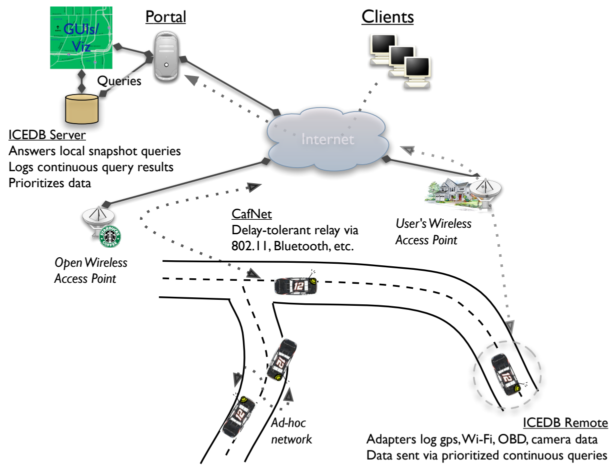

Figure 1. CarTel Architecture.

Figure 2. Screenshot of the AutoPortal prototype. Here, the user is visualizing a recently driven route.

Figure 2 show a sample visualization produced by AutoPortal.

2. *CafNet:* A networking infrastructure for carry-and-forward networks that leverages variable and intermittent network connectivity. CafNet is designed to work with a heterogeneous set of network technologies and manages the routing of data across many unreliable, high-latency links. CafNet treats the mobility of its network medium (e.g., USB keys, PDAs, cell phones) as an asset that helps it extend the reach of traditional networks.

3. *ICEDB:* A device-level data management infrastructure that collects, pre-processes, and prioritizes information on remote nodes running CarTel software. Each device is able to automatically adjust its data-collection schema depending on the sensors present in the car. Data is aggregated and queries are processed using a simple stream-processing engine.

3 Demonstration Highlights

In this demo we show a prototype of the CarTel sensor computing system, including the following highlights:

1. *Visual query system:* We demonstrate our graphical query system using historical data collected by our CarTel deployment in Boston and Seattle. This demo shows how our system can be used to compare travel times for a user's common routes. For example, one of our users commonly travels from Winchester to Boston. Each day he must select one of four routes to take into work. Using our collected data, we show how such a user can sift through the thousands of traces he has collected to meaningfully compare these four routes.

2. *Adaptive Data Prioritization:* We demonstrate our dynamic prioritization algorithm for query results using a CarTel node at the conference site. This node is preloaded with data and is using an artificially rate-limited connection to simulate scarce bandwidth available to the units in the field. Demo participants use AutoPortal to send queries to this unit and observe the order in which results come back.

3. *Opportunistic data transfers:* We rent a car and instrument it with a CarTel node in it. During the demo, this car travels around Boulder, collecting sensor data and uploading the data to our servers over wireless connections. At the conference site, we use the AutoPortal to show a trace of where the car has been traveling in the last few minutes.