CarTel: A Distributed Mobile Sensor Computing System

Bret Hull, Vladimir Bychkovsky, Yang Zhang, Kevin Chen, Michel Goraczko, Allen Miu, Eugene Shih, Hari Balakrishnan, Sam Madden

MIT CSAIL

Presented by Grigoris Karvounarakis

at CIS 700/005, April 2007

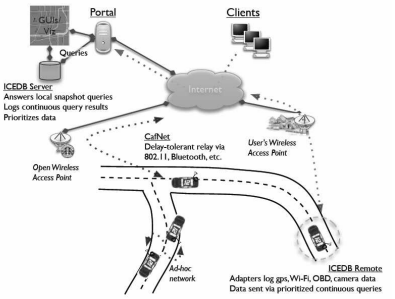
(using material from related presentations of the authors)

What is CarTel?

- Distributed software system that makes it easy to:
  - collect,
  - process,
  - deliver,
  - visualize & analyze

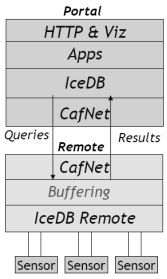
  data from *mobile* sensors (cars, phones, etc)
- Goals + Challenges:
  - Intermittent network connectivity + mobility
  - Lots of data ("media-rich" sensors)
  - Heterogeneous data
  - Programmability

CarTel System Architecture

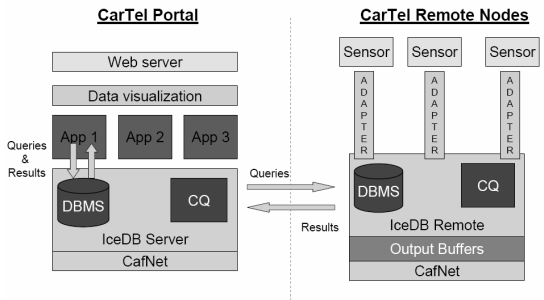


CarTel Software Components

- *CarTel Portal*: Centralized, visual user interface
- *IceDB*: Intermittently connected DB
  - Centralized declarative queries
  - Executed in distributed fashion by mobile nodes
  - Delay-tolerant continuous query processing
- *CafNet*: CarTel's network stack
  - Handles variable and intermittent connectivity



CarTel Software Architecture



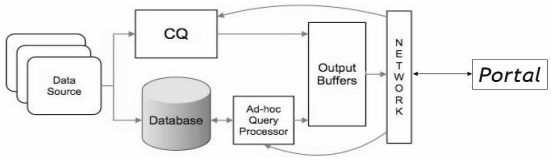
IceDB: Intermittently Connected DB

- Delay tolerant, distributed continuous query engine
  - Highly variable connectivity & bandwidth
- SQL extensions to handle intermittent connectivity
  - To prioritize results
- Adapters for managing heterogeneous data types
  - Meta-data package describing attributes ofsensor
  - Create local tables for sensor readings
  - Acquire tuples from sensor
  - Parse sensor readings

(Traditional) Continuous Queries

- Current model for stream processing
  - Process data streams via long-running queries
  - Windowed aggregates, filters, windowed joins, merges, etc.
- Network is assumed to be "always on"
  - Disconnection is a fault to be masked (or a failure occurs)

Delay-Tolerant Continuous Queries



- IceDB stages data into output buffers to hide variable connectivity
- Key idea: Data in output buffers get re-evaluated dynamically, each time a new item arrives into it

Result Prioritization

- Limited BW necessitates deliberate ordering
- Three simple SQL extensions
  - For local (per-box) ordering:
    - PRIORITY (for whole queries)
    - DELIVERY ORDER BY (within query results)
  - For global ordering (according to feedback from the portal, possibly across all sensors):
    - SUMMARIZE AS

PRIORITY

- Idea
  - Some queries are more important than others
- Details
  - Add PRIORITY clause to SQL
  - Drain output buffers in priority order

```
SELECT lat, lon, mph
FROM gps
WHERE mph > roads.mph_limit
BUFFER IN driver
PRIORITY 10
```

```
SELECT oil_temp
FROM obd
BUFFER IN engine
PRIORITY 1
```


DELIVERY ORDER BY

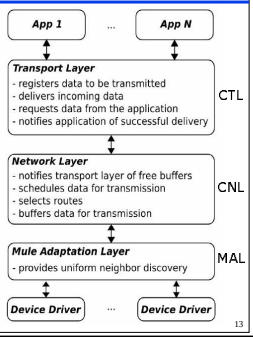
- Idea
  - Prioritize tuples within query result
- Details
  - Query specifies transmission order via DELIVERY ORDER BY clause
  - User-defined ordering function
  - Operates over entire query output buffer

SUMMARIZE

- Idea
  - Nodes send server low-resolution summary of output buffer contents
  - Server sends back transmission ordering
- Details
  - Users specify "summarization query" alongside main query
  - Server ranks segments using app-defined metric
  - Ranking pushed to nodes to set output ordering

CafNet: A Delay-Tolerant Network Stack

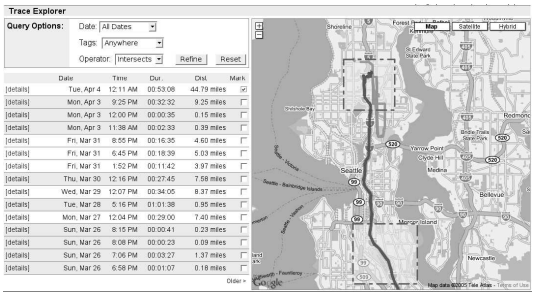
- Data moves through regions of highly variable connectivity
- "Mule" = element that stores data to be relayed toward the destination when "the time is right"
  - a delay-tolerant network (DTN)
- CafNet delivers results to portal and queries to nodes



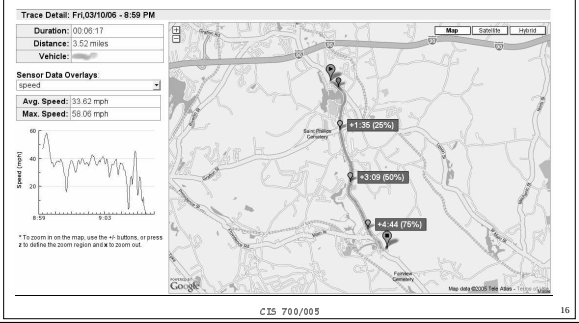
CafNet Optimizations

- Basic version: Callback + no buffering
- Problem: Connectivity may not even be long enough to package and send data
- Solution:
  - Add buffering at CNL level
  - CNL sends whatever is in the buffer as soon as connectivity is available
  - Callback when there is space in buffer
- Buffer size requirements vary (vs. dynamic priority)
  - Allow applications to set desired size

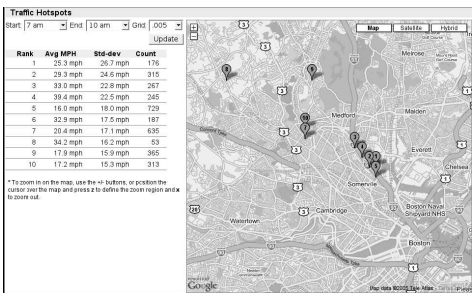
CarTel Portal: Traces and Interest Regions



CarTel Portal: a Single Trace



Library for Geographic Overlays



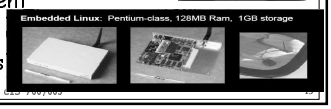
Other vehicle-based Applications

- Smart route finding & congestion mgmt
  - Past + current data
- Fleet mgmt/automotive diagnostics
  - E.g., trucks, taxis, buses
- Visual mapping (images, video) of regions
  - Pictures for driving directions
  - Surveillance videos
- Civil and environmental monitoring
  - E.g., to measure pollution or potholes
- Wireless network monitoring
  - "can you hear me now"

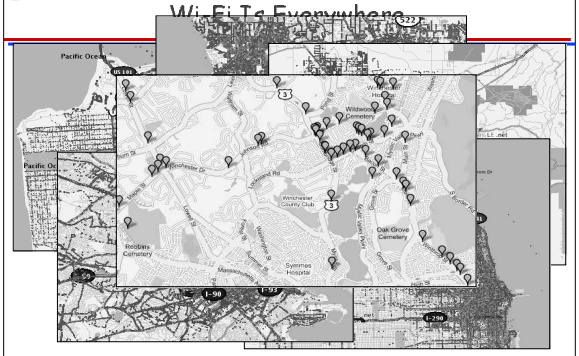
Experimental Setup

- 6 cars equipped with CarTel box and software
  - Driving normally in parts of the Boston area
  - ~32K access points (APs) mapped in all on a relatively small number of distinct routes
  - ~300 drive hours
- Fast scanning of WiFi access points, caching of AP parameters to speed up connection establishment
- 25sec connections
- Median upload: 30KB/s



Wi-Fi Is Everywhere



Images from WiGLE.net and CarTel [20]

Conclusions

- Mobile sensor networks can sense at much higher scale over large areas than static networks
- Several applications: traffic, fleet management, automotive diagnostics, wireless network monitoring, civil/environmental monitoring, …
- Key challenges: heterogeneous data, intermittent connectivity, programmability, privacy
- In urban areas, Wi-Fi is a viable uplink technology
  - Legal/privacy issues?
  - Cheaper than using cell-based?
  - They also discuss using CarTel on cell phones