Respawn: A Distributed Multi-Resolution Time-Series Datastore

Maxim Buevich, Anne Wright, Randy Sargent, Anthony Rowe

Electrical & Computer Engineering, Robotics Institute
Carnegie Mellon University

IEEE Real-Time Systems Symposium (RTSS) 2013
Scenario
Motivation

- **Buildings -> Smart Buildings**
  - (20,000+ sensing points on CMU Pittsburgh campus)

- **Applications**
  - Structural health monitoring
  - Smart grid
  - Environmental sensing
  - Geo-physical monitoring
Problem Space

• Enable interactive querying and visualization ("browsing") of time-series sensor data.

• Goal: Serve data from many intermittently-connected edge nodes while transparently masking latency for clients.

• Challenges
  • Many physically-distributed data sources
  • High-frequency data
  • Unreliable networks
  • Embedded platforms
Current Approaches

- Centralized Storage
  - SQL: MySQL, SQLite
  - NoSQL: MongoDB, CouchDB

- Distributed Storage
  - KV: Hbase, Hypertable
  - Time-series: OpenTSDB

- In-Network Storage
  - Lance
  - TAG
Respawn Approach

- **Key techniques:**
  - multi-resolution tiling
  - cloud-to-edge partitioning
Outline

• Problem/Approach
• Design/Implementation
  • High-Level Architecture
  • Multi-Resolution Storage
  • Data Migration
  • Request Handling
• Evaluation
• Conclusion
High-Level Architecture

- **Edge Nodes**: Embedded storage device near source
- **Cloud Node** (back-end): Fast persistent storage on server
- **Dispatcher** (front-end): Fast client redirection to edge/cloud
- **Protocols**: XMPP (events), HTTP (data)
Target Platform

- Environmental Sensor ($20)
  - 16 MHz 8-bit ATmega
  - 16 KB SRAM
  - 128 KB flash

- Olimex iMX233 ($60)
  - 450 MHz ARM9
  - 64 MB RAM
  - 64GB uSD flash

- Intel Xeon Machine ($1000+)
  - 1.8GHz 8-core
  - 4GB DRAM
  - 1TB+ HDD
Multi-Resolution Data Storage

- Open-source BodyTrack Dataset (BTDS)
- **datapoint** = (time, value, count, stddev)
Multi-Resolution Data Storage

- Open-source BodyTrack Datastore (BTDS)
- **datapoint**: (time, value, count, stddev)

**tile size:**

\[ k = 4 \]

**tile request:**

\[ \log_2(\text{period}) \cdot \frac{\text{time}}{k \cdot \text{period}} \]

getTile(level, offset)

**example tile request:**

\[ \log_2(0.5) \cdot \frac{8}{4 \cdot 0.5} \]

getTile(-1, 4)
Data Migration

- Edge nodes notify of events via XMPP
- Data/metadata fetched via HTTP
- “Periodic” migration and “Proactive” migration
Request Handling

- Dispatcher redirects client requests to edge/cloud.
- **REQUEST**: (device, channel, level, offset)
  - “HTTP/1.1 GET /tile/sensor.temperature/10.2609.json”
- **RESPONSE**: JSON object
Outline

- Problem/Approach
- Design/Implementation
- Evaluation
  - Multi-resolution Storage
  - Request Redirection
  - Data Migration
- Conclusion
Evaluation: Multi-Res Storage

- Query/ingest performance versus total datastore size
- Respawn compared with MySQL, SQLite, and OpenTSDB

![Graphs showing query and ingest performance](image)

Dataset Size (# of datapoints)

1K-point range query time (ms)

- Respawn
- SQLite
- MySQL
- OpenTSDB

Stored Data Size (number of raw datapoints)

1K-point Ingest Time (seconds)

- Respawn (smoothed)
- MySQL (smoothed)
- SQLite (smoothed)
- OpenTSDB (smoothed)
Evaluation: Multi-Res Buffering

- Store ingest performance benefits from buffering.
Evaluation: Request Handling

- Experimentation with central node and httpperf workloads
Evaluation: Migration

- Trace-driven from user study
- Average latencies for 5% migration limit for tiles:

  - **no migration**: 350 ms
  - **“periodic” migration**: 180 ms
  - **“proactive” migration**: 130 ms
  - **“periodic” + “proactive”**: 80 ms
Deployments:

- Pittsburgh – CMU Scaife Hall
  - 100+ sensor feeds
- Les Anglais, Haiti – Rural microgrid
  - 52-home deployment, 1 high-frequency three-phase meter
Summary

- Multi-resolution data storage for interactive visualization and range-based queries
  - Masks time-series data query
  - Data migration significantly improves client latencies
  - Copes with network outages
  - Request redirection approach enables a single Respawn dispatcher to serve over 15K clients
Thank You!