# GHT: A Geographic Hash Table for Data-Centric Storage

Sylvia Ratnasamy, Brad Karp, Li Yin, Fang Yu, Deborah Estrin,
Ramesh Govindan, Scott Shenker (UC Berkeley, UCLA, USC)

Presentation by Qing Fang for CS428

# This Paper ...

## Outline three canonical data dissemination methods

- External storage
- Local storage
- Data centric storage (DCS)
  - Sensed data are stored at a node determined by the name associated with the sensed data

## Serves two aims

- Identify the circumstances where DCS is the preferred dissemination method
- Present design criteria for scalable, robust DCS, and a DCS system that meets those criteria GHT

  Evaluate GHT as a DCS system via simulation

# Assumptions

- A class of sensornets that is most relevant to data dissemination
- Large-scale sensornets with nodes spread out over an area whose approximate geographic boundaries are known to the the network operators
- Nodes know their geographic location
- The sensornet is connected to the outside world through a small number of access points

# Metrics

- Total usage – the total number of packets sent in the sensornet
- Hotspot usage – the maximal number of packets sent by any particular sensornet node

# DCS is preferable in cases where:

- The sensornet is large compared to the number of events (otherwise **local storage** may be preferable)
- There are many detected events and not all event types are queried (otherwise **external storage** may be preferable)

Especially true when summaries are used in queries.

# Design Criteria for Scalable Robust DCS

- **Persistence**: a (*k, v*) pair stored in the system must remain available to queries, despite sensor node failures and changes in the sensor network topology
- **Consistency**: a query for k must be routed correctly to a node where (k, v) pairs are currently stored; if this node changes, queries and stored data must choose a new node consistently
- **Scaling in database size**: storage should not concentrate at any one node
- **Scaling in node count**: as the number of nodes increases, the system's total storage capacity should increase, and the communication cost of the system should not grow unduly. Nor should any node become a concentration point of communication
- **Topological generality**: should work well on a broad range of network topologies

# Geographic Hash Table (GHT)

- GHT hashes keys into geographic coordinates, and stores a key-value pair at the sensor node geographically nearest the hash of its key.
- The system replicates stored data locally to ensure persistence when nodes fail.
- A data object is associated with a key and each node in the system is responsible for storing a certain range of keys.
- A name-based routing algorithm allows any node in the system to locate the storage node for an arbitrary key.
- This enables nodes to *put* and *get* files based on their key, thereby supporting a hash-table-like interface
- GHT uses the GPSR geographic routing algorithm as the underlying routing system

# Algorithm

- GPSR

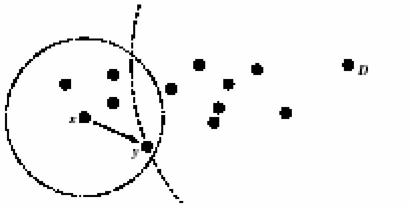


Figure 1: Greedy Forwarding Example: $x$ forwards to $y$, its neighbor closest to $D$.

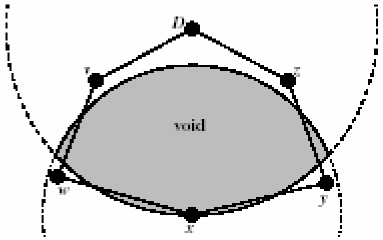


Figure 2: Void Example: $x$ has no neighbor closer to $D$.

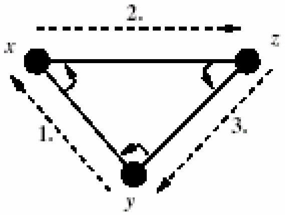


Figure 3: Right-hand Rule Example: Packets travel clockwise around the enclosed region.

- The home node and home perimeter

  Because a GHT packet is not addressed to a specific node, but only to a specific location, the packet enters perimeter mode at the home node. The packet then traverses the entire perimeter that encloses the destination, before returning to the home node and being consumed.

# DCS design criteria for scalability and robustness

1. Perimeter refresh protocol

- Replicates stored data for key *k* at nodes around the location to which *k* hashes, and ensures that one node is chosen consistently as the *home node* for that *k* – consistency & persistence
- Typically uses highly local communication – scalability in network size
- By hashing keys, GHT spreads storage and communication load between different keys evenly throughout the sensornet – scalability in database size

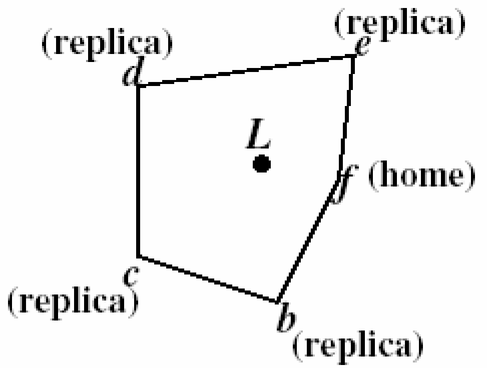


Figure 6: Node *f* becomes the new home node, and recruits replicas *b*, *c*, *d*, and *e*.

2. Structured replication

Avoids creating a hotspot of communication and storage at their shared home node by employing *structured replication* – scalability in database size

# Algorithm (cont'd)

- Perimeter refresh protocol (PRP) – ensure persistence and consistency

  To accomplish replication of key-value pairs and their consistent placement at the appropriated home nodes when the network topology changes.

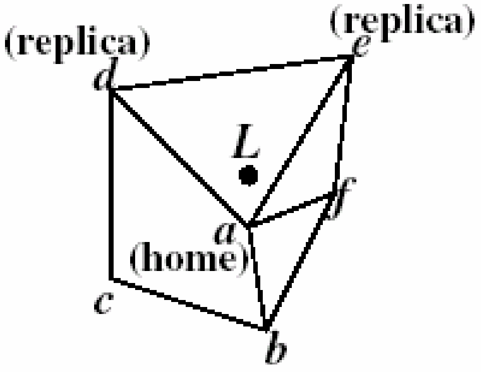


Figure 4: Key stored at location $L$, home node $a$, replicas $d$ and $e$ on the home perimeter.

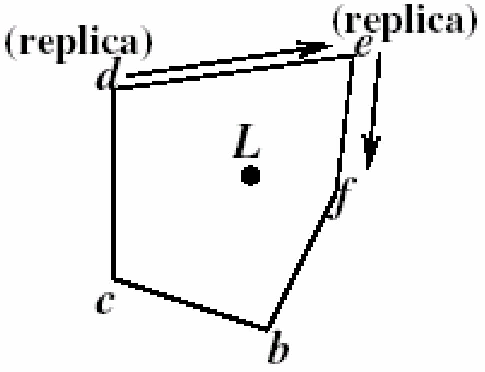


Figure 5: Time $T_l$ after node $a$ fails, replica $d$ initiates a refresh for $L$.

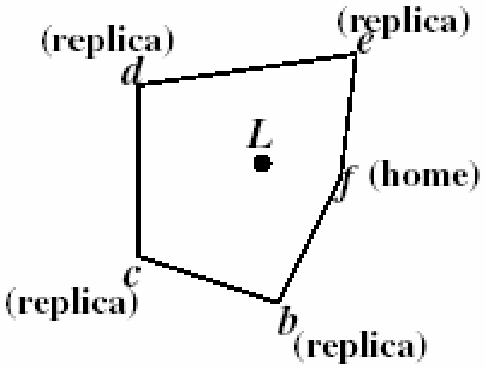


Figure 6: Node $f$ becomes the new home node, and recruits replicas $b$, $c$, $d$, and $e$.

# Three Timers

- Refresh timer
- Takeover timer
- Death timer

# Algorithm (cont’d)

## Structured replication

- If too many events with the same key are detected, that key’s home node could become a hotspot, both for communication and storage.
- Use a hierarchical decomposition of the key space

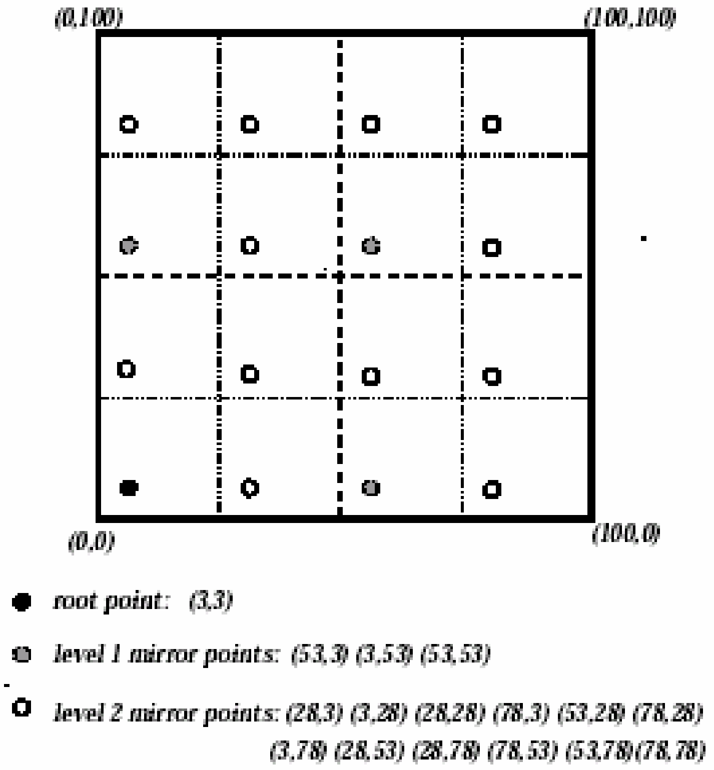

(0,100)
(100,100)
(0,0)
(100,0)
● root point: (3,3)
◉ level 1 mirror points: (53,3) (3,53) (53,53)
○ level 2 mirror points: (28,3) (3,28) (28,28) (78,3) (53,28) (78,28) (3,78) (28,53) (28,78) (78,53) (53,78)(78,78)

Figure 7: Example of Structured Replication with a 2-level decomposition.

# Simulations

Small-scale networks, wireless details (ns-2, less than 100 nodes)

- Stable and static nodes
- Static but failing nodes
- Stable but mobile nodes

Large-scale networks, no wireless details, no PRP (self built simulator, 100,000 nodes)

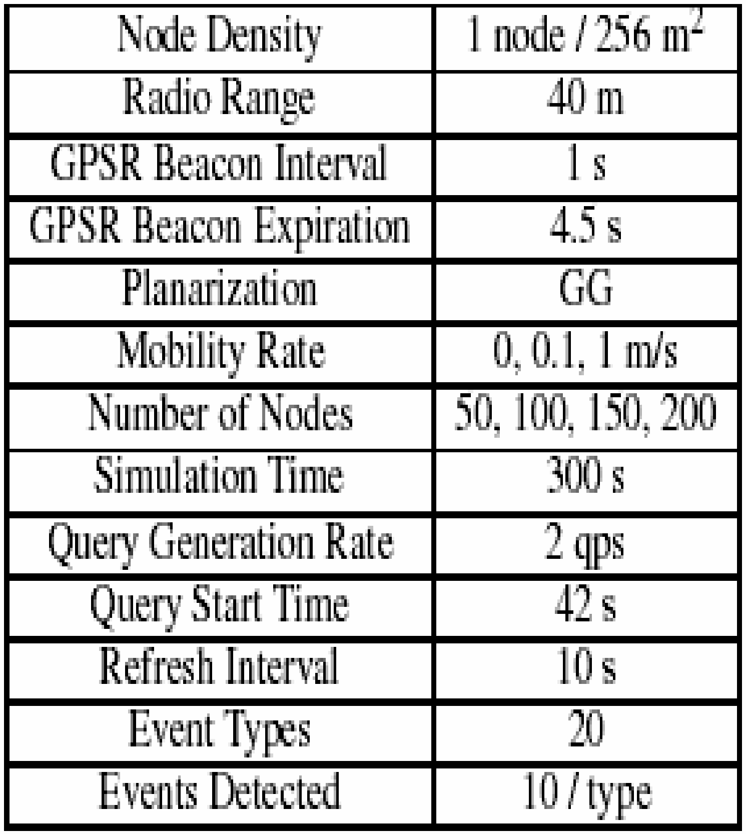
# Small Network Simulations Using NS-2

Setup

- Environmental noise and propagation obstacles are not considered
- Constant node density
- Single query node

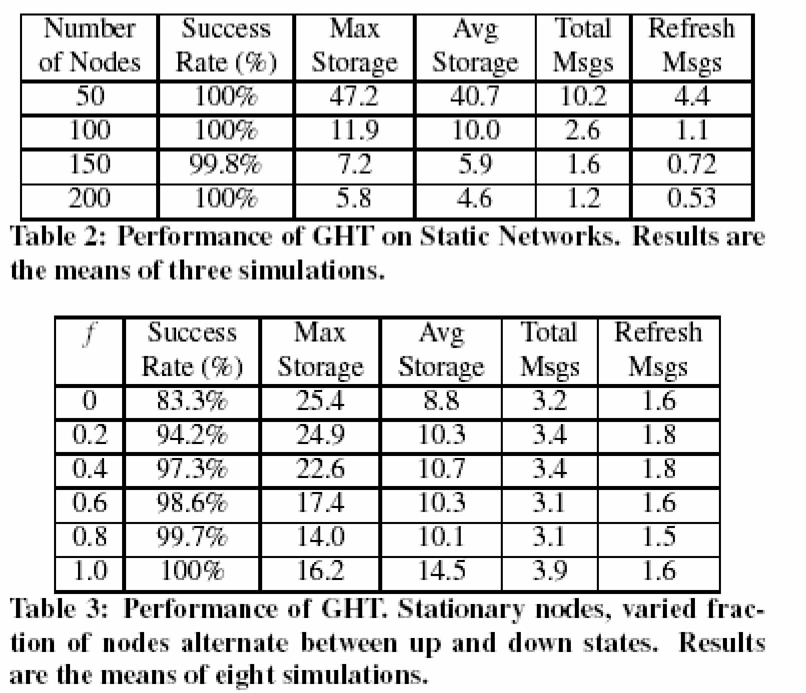
Performance Metrics

- Availability of data
- Load placed on nodes, both in communication and storage of events



| | |
|---|---|
| Node Density | 1 node / 256 $m^2$ |
| Radio Range | 40 m |
| GPSR Beacon Interval | 1 s |
| GPSR Beacon Expiration | 4.5 s |
| Planarization | GG |
| Mobility Rate | 0, 0.1, 1 m/s |
| Number of Nodes | 50, 100, 150, 200 |
| Simulation Time | 300 s |
| Query Generation Rate | 2 qps |
| Query Start Time | 42 s |
| Refresh Interval | 10 s |
| Event Types | 20 |
| Events Detected | 10 / type |

**Table 1: GHT simulation parameters in *ns*-2 simulations.**

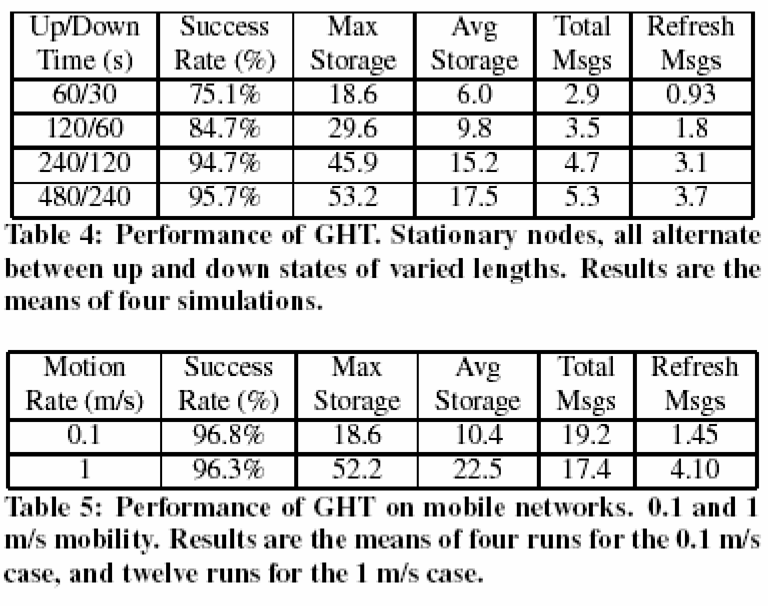


| Number of Nodes | Success Rate (%) | Max Storage | Avg Storage | Total Msgs | Refresh Msgs |
|---|---|---|---|---|---|
| 50 | 100% | 47.2 | 40.7 | 10.2 | 4.4 |
| 100 | 100% | 11.9 | 10.0 | 2.6 | 1.1 |
| 150 | 99.8% | 7.2 | 5.9 | 1.6 | 0.72 |
| 200 | 100% | 5.8 | 4.6 | 1.2 | 0.53 |

**Table 2: Performance of GHT on Static Networks. Results are the means of three simulations.**

| $f$ | Success Rate (%) | Max Storage | Avg Storage | Total Msgs | Refresh Msgs |
|---|---|---|---|---|---|
| 0 | 83.3% | 25.4 | 8.8 | 3.2 | 1.6 |
| 0.2 | 94.2% | 24.9 | 10.3 | 3.4 | 1.8 |
| 0.4 | 97.3% | 22.6 | 10.7 | 3.4 | 1.8 |
| 0.6 | 98.6% | 17.4 | 10.3 | 3.1 | 1.6 |
| 0.8 | 99.7% | 14.0 | 10.1 | 3.1 | 1.5 |
| 1.0 | 100% | 16.2 | 14.5 | 3.9 | 1.6 |

**Table 3: Performance of GHT. Stationary nodes, varied fraction of nodes alternate between up and down states. Results are the means of eight simulations.**



| Up/Down Time (s) | Success Rate (%) | Max Storage | Avg Storage | Total Msgs | Refresh Msgs |
|---|---|---|---|---|---|
| 60/30 | 75.1% | 18.6 | 6.0 | 2.9 | 0.93 |
| 120/60 | 84.7% | 29.6 | 9.8 | 3.5 | 1.8 |
| 240/120 | 94.7% | 45.9 | 15.2 | 4.7 | 3.1 |
| 480/240 | 95.7% | 53.2 | 17.5 | 5.3 | 3.7 |

**Table 4: Performance of GHT. Stationary nodes, all alternate between up and down states of varied lengths. Results are the means of four simulations.**

| Motion Rate (m/s) | Success Rate (%) | Max Storage | Avg Storage | Total Msgs | Refresh Msgs |
|---|---|---|---|---|---|
| 0.1 | 96.8% | 18.6 | 10.4 | 19.2 | 1.45 |
| 1 | 96.3% | 52.2 | 22.5 | 17.4 | 4.10 |

**Table 5: Performance of GHT on mobile networks. 0.1 and 1 m/s mobility. Results are the means of four runs for the 0.1 m/s case, and twelve runs for the 1 m/s case.**

Limitations: Localized replication of this form is of little use if all the nodes in an area fail at the same time. Can be remedied by using multiple hash functions.