CENS Center for Embedded Networked Sensing

Seismic Array Software System

Sam Irvine, Martin Lukac, Andrew Parker, Allen Husker, Igor Stubailo,
Richard Guy, Paul Davis, Deborah Estrin
CENS System Lab – http://research.cens.ucla.edu

Motivation: Long-term Deployment of a Portable Broadband Seismic Array

About
- Part of the Middle America Subduction Experiment (MASE)
- Partnered with Caltech and UNAM

Goals
- Map the subducted slab beneath Mexico
- Examine slow earthquakes observed at this subduction zone
- Examine volcanic earthquakes observed at this subduction zone
- Study the propagation of seismic waves in Mexico City

Needs
- Line of seismic station: Acapulco to Tampico through Mexico City
- 100 Stations total, 5-20 km apart
- 100 Hz broadband seismometers

The Setup: High Powered 802.11b Connect 50 Stations to Mexico City

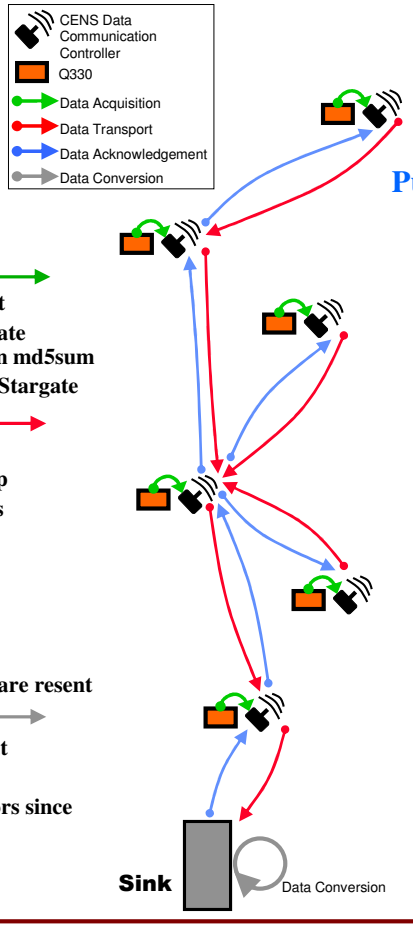
Physical Topology Characteristics
- Non-uniformity in Topology
- Variable Spacing: many factors in choosing a site
  - Terrain and vegetation
  - Policy – need local permission for each site
  - Cable length and antenna height
  - Seismic Noise
- Distance between stations: 100m to 20km
  - Relays fill in critical gaps
  - Some stations have internet connections and hard drives
- Network topology is the physical topology
- Each node only has a single downstream neighbor
- Not completely linear – local clusters and star topology
- Max hops is 15 - largest cluster of nodes is 20
- End-to-end connections are unreliable, unstable, and slow
- Data is multi-hoped delivered to a sink
- Need EVERY bit – cannot lose any data

- CDCC: Stargate SMC 802.11b ad-hoc 1-4GB CF Cards

Duiker: Software for end to end system autonomous seismic data collection

Duiker
- Data acquisition tools for the Q330
  - Runs in linux
  - Small code, memory, and CPU footprint
- Why not Antelope?
  - Cost!
  - Difficult to use
  - Not suited for small embedded platform

Duiker components
- Acquisition: runs on microservers
  - Collects data from Q330 over ethernet
  - Bundles contain raw Q330 packets, state information configuration info, and an md5sum
  - < 1% of CPU and < 1MB of RAM on Stargate
- Transport: runs on microservers
  - Moves bundles hop-by-hop to a sink
  - Bundles transferred over tcp using scp
  - Storage priority given to local bundles
- Data Acknowledgement
  - Initiated at the sink
  - List of received files at sink updated throughout network
  - Each node uses list to delete files
  - Old local bundles that are not ack'ed are resent
- Data Conversion runs at the sink
  - Converts bundles into *miniseed* format
  - No conversion happens on Stargate
  - Allows recovery from conversion errors since all raw packets from Q330 are saved



October 2005 Status
- 40 of 50 sites completed
- Additional relays required to connect paths to sinks
- Duiker completed
  - Instrumentation underway

Purposed Measurement Instrumentation
- Transport component will keep track of:
  - When it first tried to send a bundle
  - Each time it begins to send the same bundle
  - When it successfully completes sending a bundle
  - The disk space used on the node on send and receive
- Compare with simulation and testbed results

Storage Estimates
- Data generated at 1-3MB per hour
- 1 GB CF card: 14 days worth of data from a single node at 3MB per hour
- For a 12 node path: 27 hours of data / 1 GB

Minimum Bandwidth Estimations
- Assume worst-case:
  - 3MB per hour = 6667 bits per second per node
  - Last hop connection to sink requires most bandwidth
- For 12 nodes, ~ 80-kbps at last hop:
  - 6,667 bps/node * 12 nodes = 80,004 bps

Latency Measurements
- Latency will be measured through simulation
- Instrumentation will report actual latency
- Depends on node uptime
  - Nodes going down means data can get lost
  - CF Cards fill up and data is delayed or lost