



Colocating sensors: impacts on infrastructure and performance

NetOps X

4 November 2019

**Session facilitators: Brian Shiro (bshiro@usgs.gov) and
Peggy Hellweg (hellweg@berkeley.edu)**

Overview

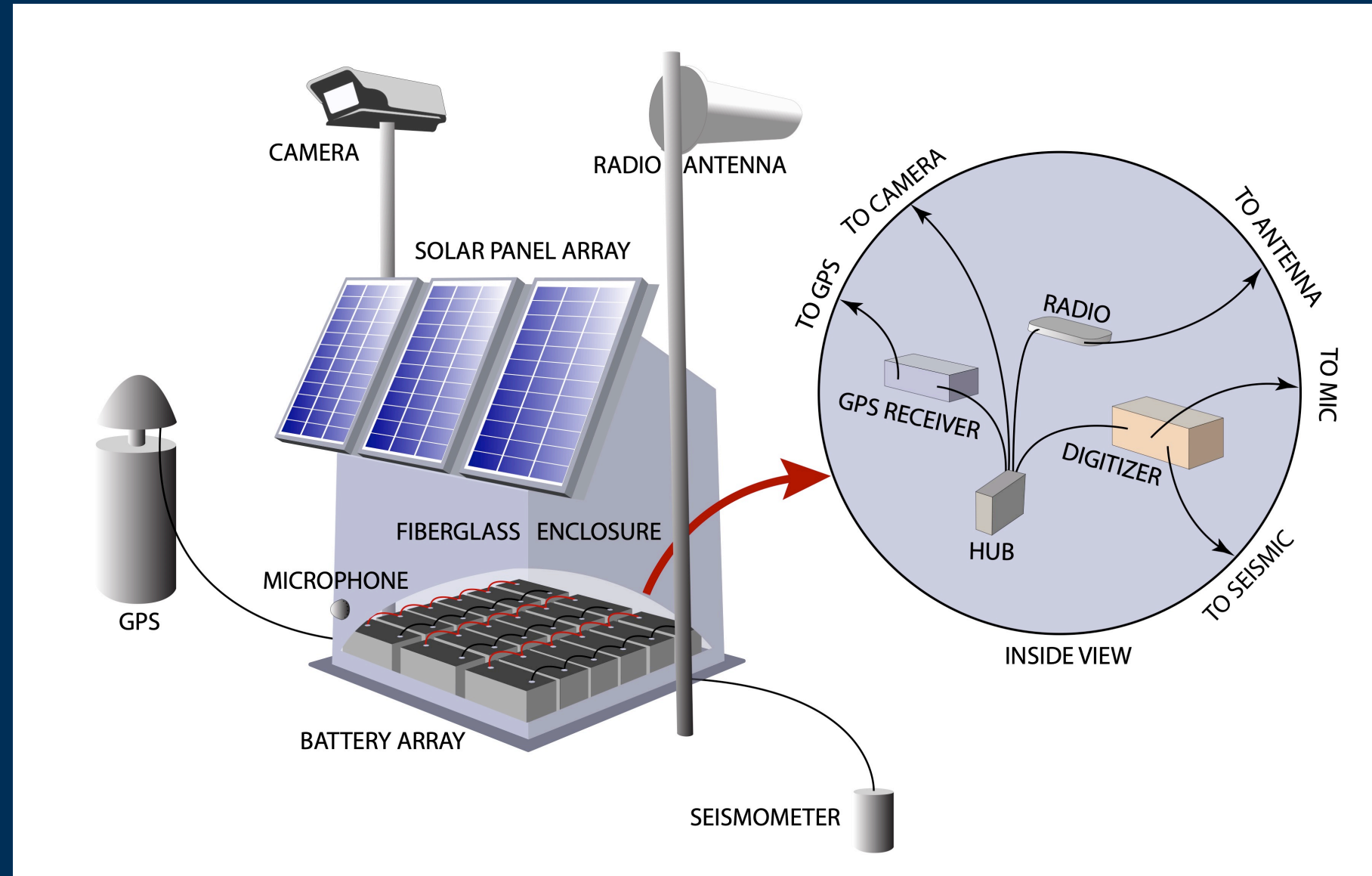
- 3 elements to this session:
 - 1) Sites with colocated sensors
 - 2) Infrastructure requirements
 - 3) Performance considerations
- Some examples of sensors that may be colocated:
seismic, GNSS/GPS, infrasound, magnetotelluric, other EM,
strain, weather, camera, tilt, gas, gravity, sea level

Questions/topics posed to the NetOps group

- How have you been able to integrate multidisciplinary data at sites?
- How do you address metadata issues using SIS or other tools?
- Do you collect locally digitized data using separate dataloggers or treat it as extra channels using seismic dataloggers?
- Have you had to adjust telemetry and power system approaches due to extra bandwidth and power required from additional sensors?
- How do you monitor state of health and performance of colocated sensors?

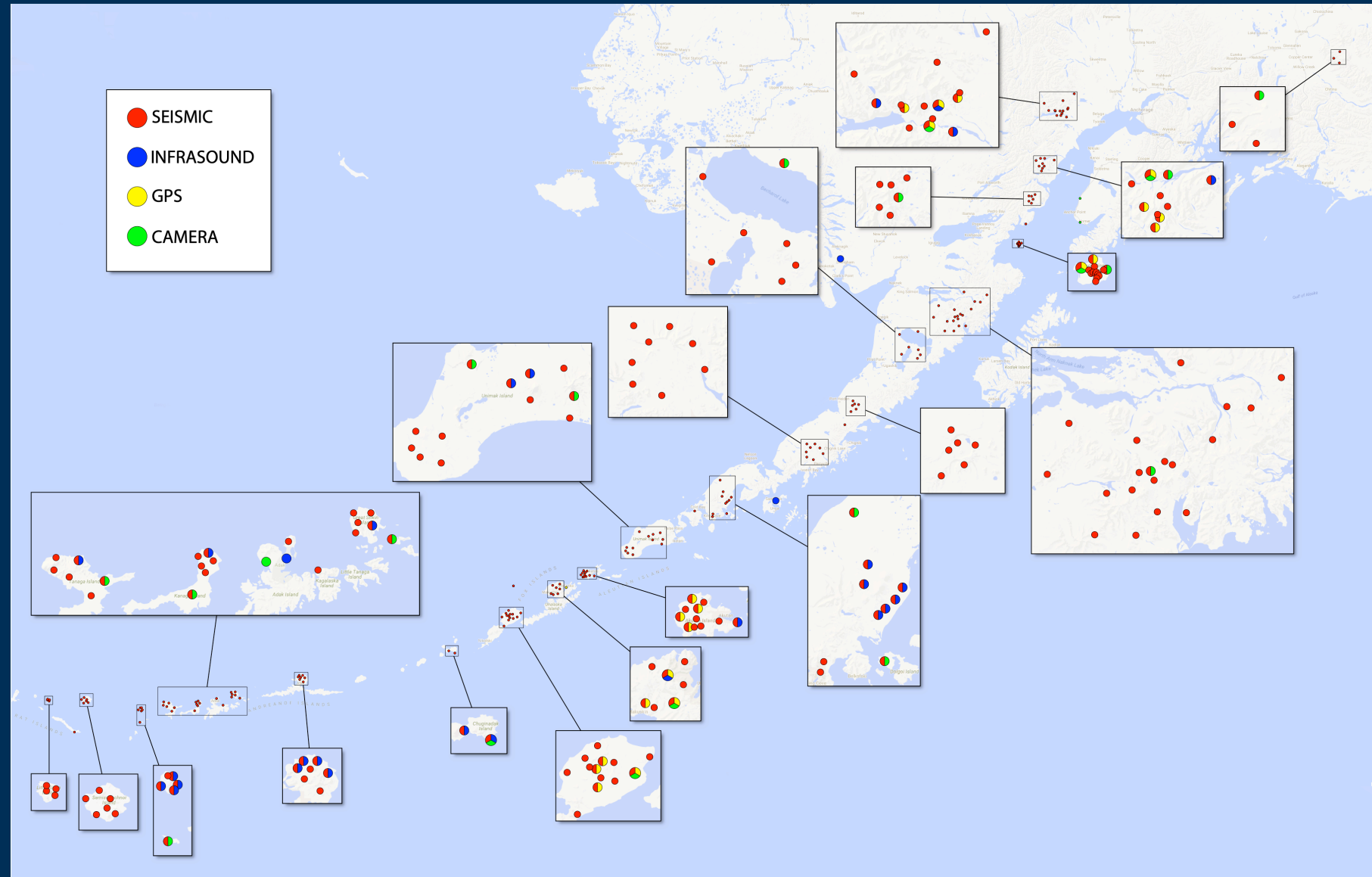
AVO

- Sites typically have seismic, GPS, camera, infrasound, and sometimes tilt
- Shared power, telemetry, and enclosure.
- Challenges relating to network traffic, power, and metadata management.



AVO

- Sites typically have seismic, GPS, camera, infrasound, and sometimes tilt
- Shared power, telemetry, and enclosure.
- Challenges relating to network traffic, power, and metadata management.



AVO

- Multi-agency partners in Alaska: AVO, AEC, TA, NTWC. Can lead to cost-sharing and permitting advantages. Group responsible for power/telemetry is typically in charge of SOH monitoring, and owner of equipment usually manages metadata.
- Site selection is often based on install criteria for a specific instrument, may not be good for another. Keep multi-purpose in mind during site recon to save problems later.
- Most colocations work out fine. Adding sensors to a site usually works out but can have significant impacts on power and telemetry.



2 separate power systems in 1 hut (poor integration)

CVO

- Nearly all CVO sites are permitted for both seismic and GPS, but almost no sites are great for both (one or the other). Hut enclosures chosen to be able to house enough solar panels and batteries for both.
- GPS data transmission issues. Daily downloads can bog down RF links. Future of BINEX questionable. How to stream data continuously and fill gaps with 'rsync' still experimental.
- GPS metadata is a huge problem. IGS is not machine readable. Newer XML variants haven't caught on. SIS developments may address some of this.
- Some sites have Axis webcams, but there are many issues with RF links b/c ethernet packets are buggy and huge.
- Infrasound is fairly easy – most issues are due to pin out wiring.

BDSN (Berkeley)

- Colocated sensors include:
GNSS (NetRS, Topcon, Septentrio (including PPP)), downhole and uphole seismic sensors, strain, pore pressure, electric and magnetic field measurements, pressure, temperature, humidity.
- Depending on the location and the instrumentation, we may have separate data loggers (originally Q680s, 980s or 1280s) or Q330s with QEP packages (nowadays)
- Determining whether the sensors all work is sometimes a challenge.

PRSN (Puerto Rico)

- Seismic + GNSS/GPS sites
- Trimble NetRs and NetR9 data loggers with Topcon receivers
- Data is recorded in data server (Ntrip BKG caster).
- Telemetry: radio (900 MHz, 2.4GHz), satellite, cell
- MySQL server to keep track of the equipment that is installed at the seismic stations and keep a web page log for all field actions taken at stations
- Currently using PDCC to generate seismic metadata (plan to use SIS in the future)
- State of health information not transmitted due to bandwidth.

Other input from the room

- Met data needed to augment other data: rain+tilt, wind+gas
- Wind to help evaluate seismic data quality, radiometer to help quantify solar input
- Cameras – remotely turn on/off, change transmission settings
- Need for good documentation since different techs might be maintaining different sensors

Summary and Conclusions