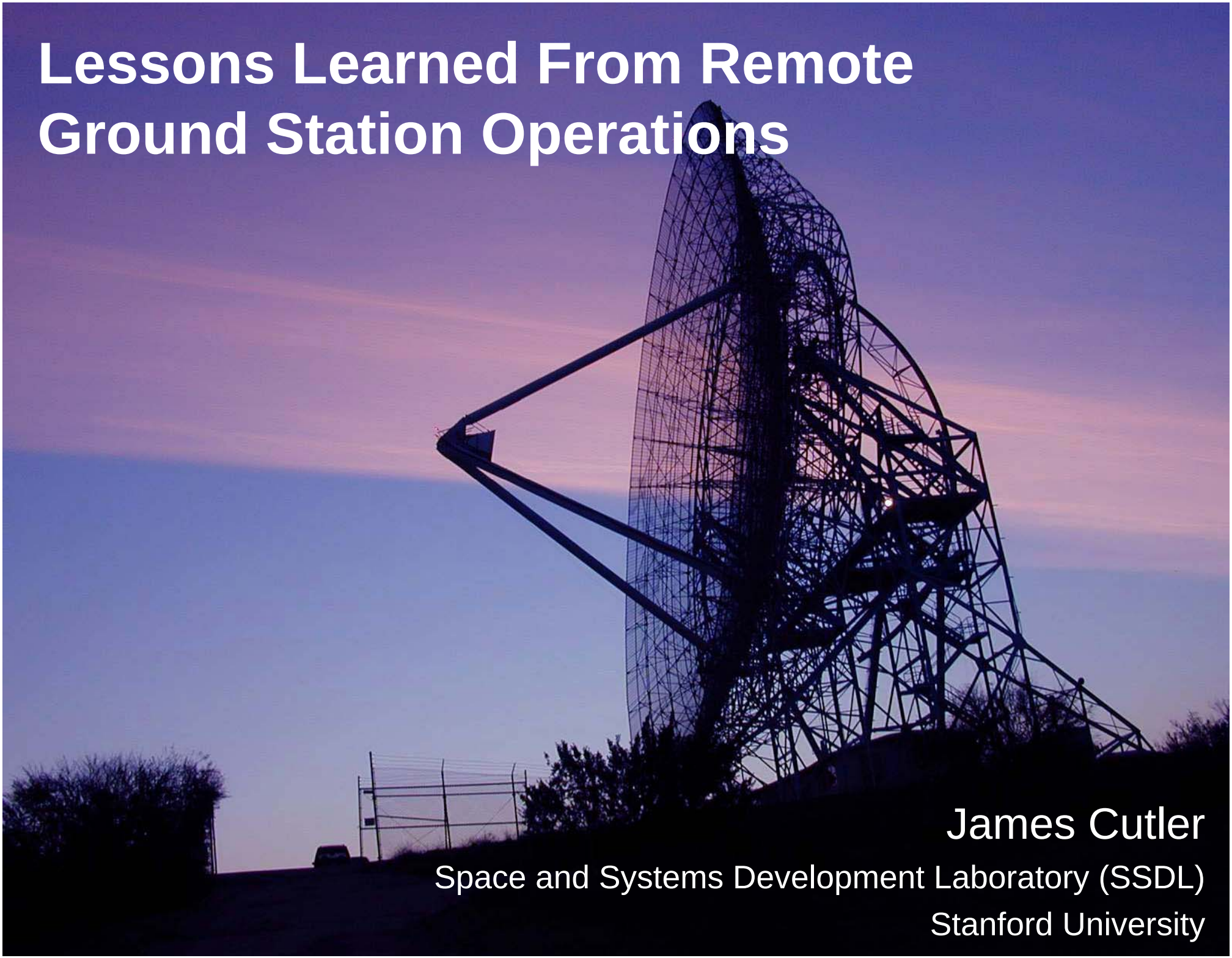
Lessons Learned From Remote Ground Station Operations



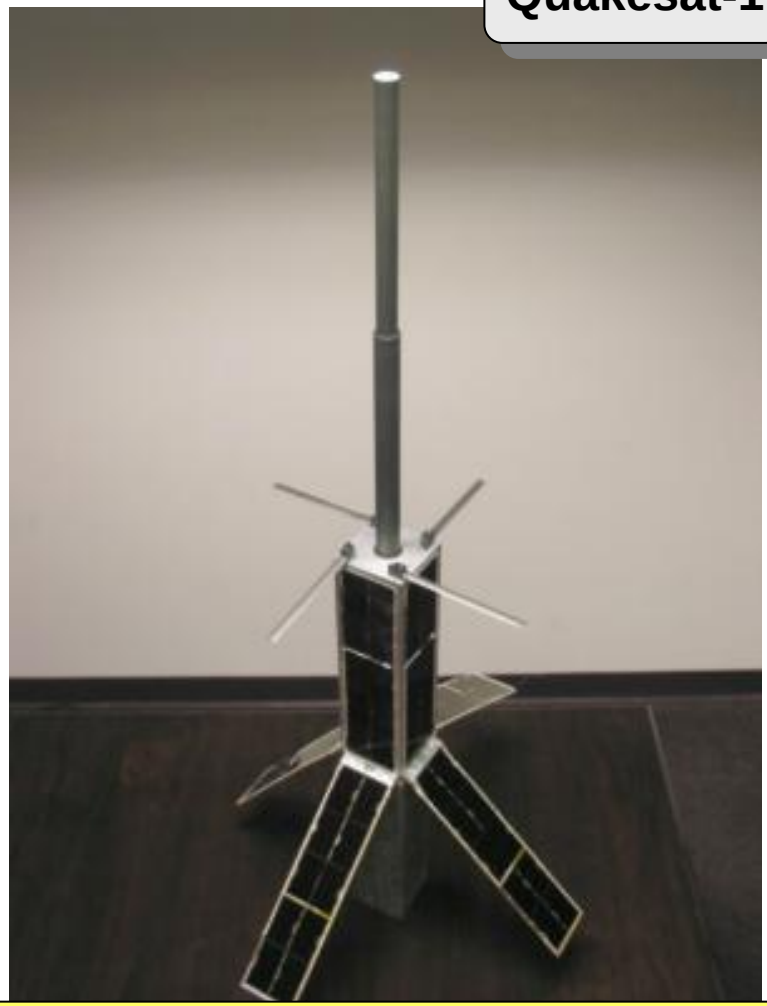
James Cutler
Space and Systems Development Laboratory (SSDL)
Stanford University

Can we talk to our satellites like we talk to Google?

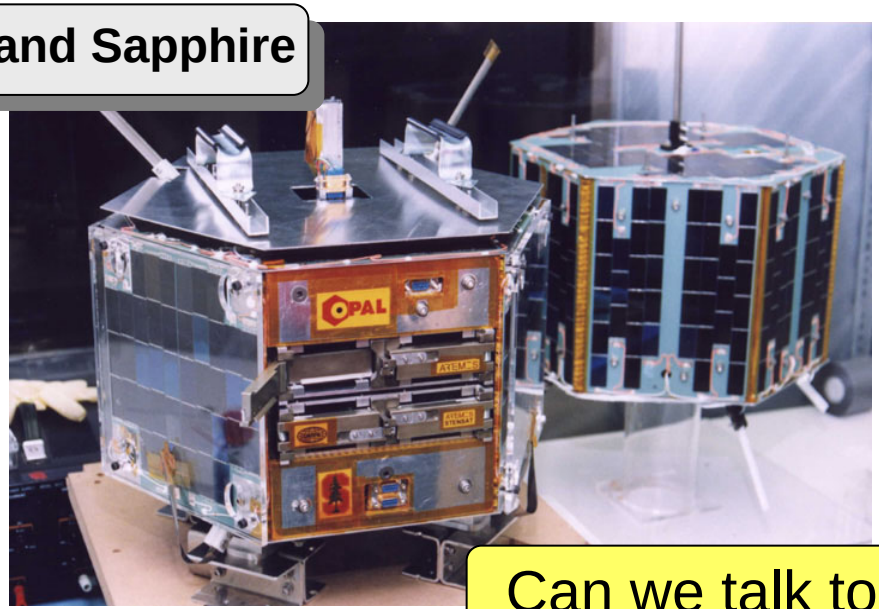
SSDL – Remote Ops Since 2000

- Five satellites in orbit
  - Opal, Sapphire, Quakesat,
  - GENESAT, MAST
  - Future…solar sails!
- Near space launches
- Earth stations:
  - Alaska (U/VHF), California (U/VHF,S)

Quakesat-1



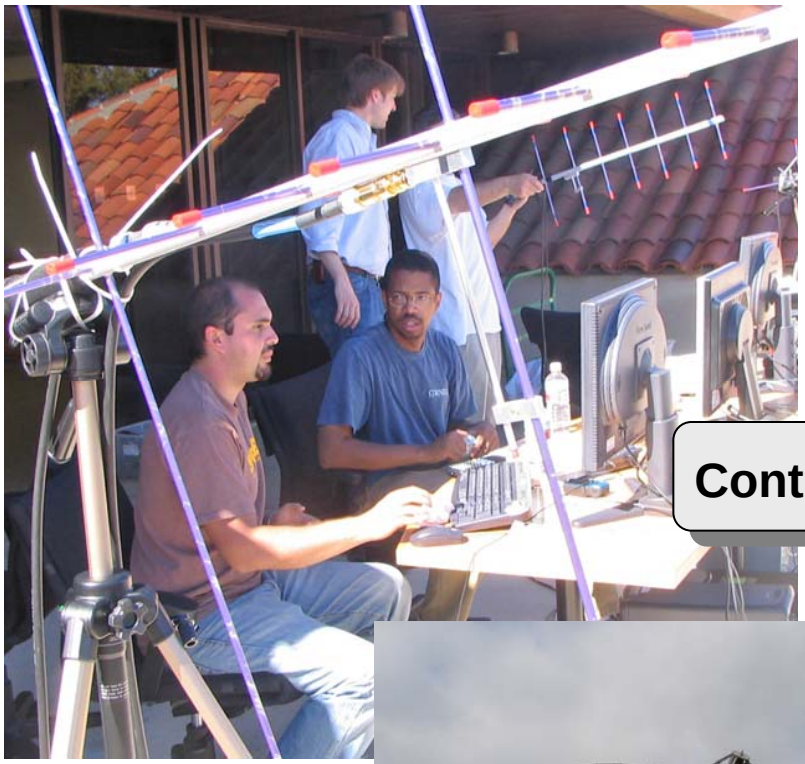
Opal and Sapphire



Can we talk to our sateljlites like we talk to Google?

High Altitude Balloon – BioLaunch B07A

Mobile S-Band at launch

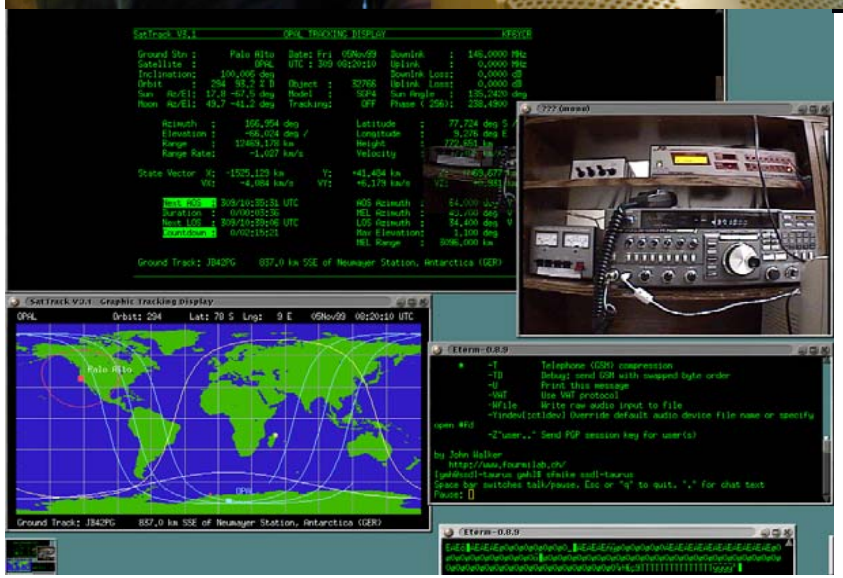


Control center

6m dish in foothills behind SU

Santa Cruz from 70K Feet

Remote Operation – Lazy or Useful?



- DNEPR 2 passes
  - ~12 @ Fairbanks, Alaska
  - ~ 4 @ Stanford, California

Lesson: Local Operation is Most Informative

- Direct access to equipment
- Higher bandwidth telemetry
  - Audio
  - Spectrum analyzer
- Tracking Roles
  - Tracking
  - Spectrum/Signal monitoring
  - Sat ops
  - Coordination Communication

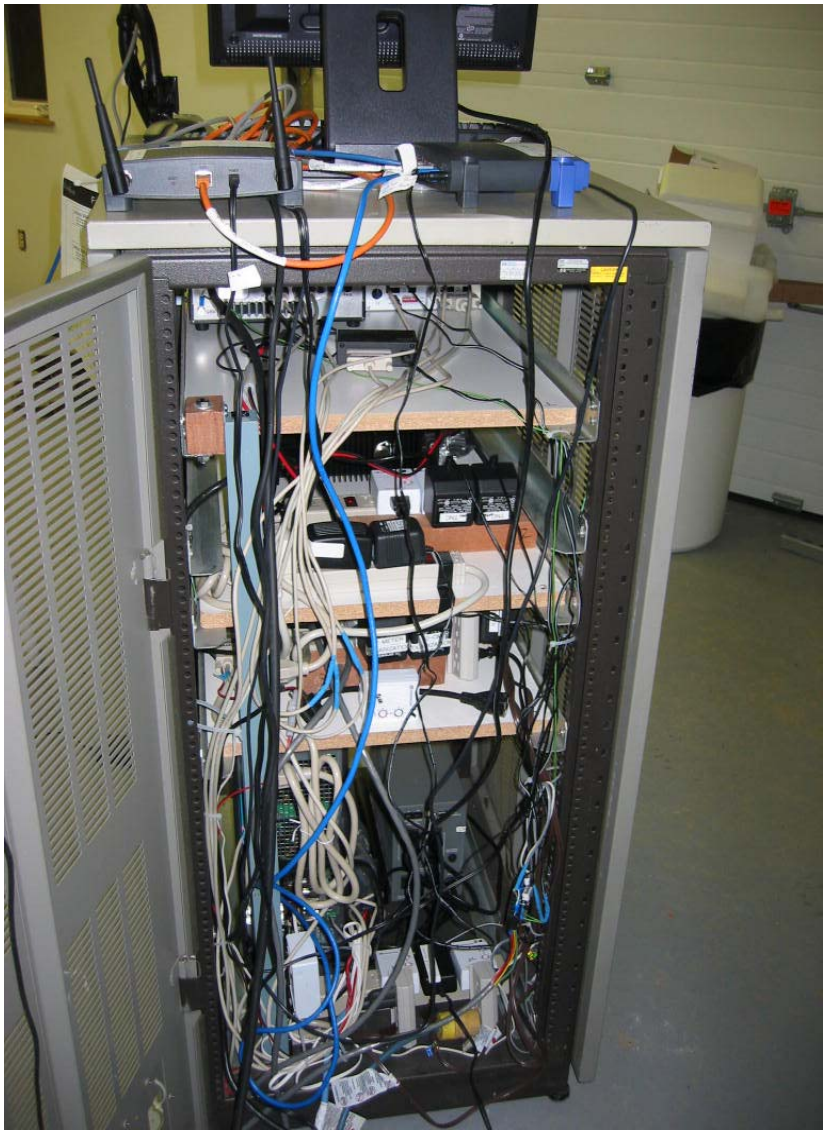
How can we improve remote operation?

Alaska GS

Lesson: IP Everywhere

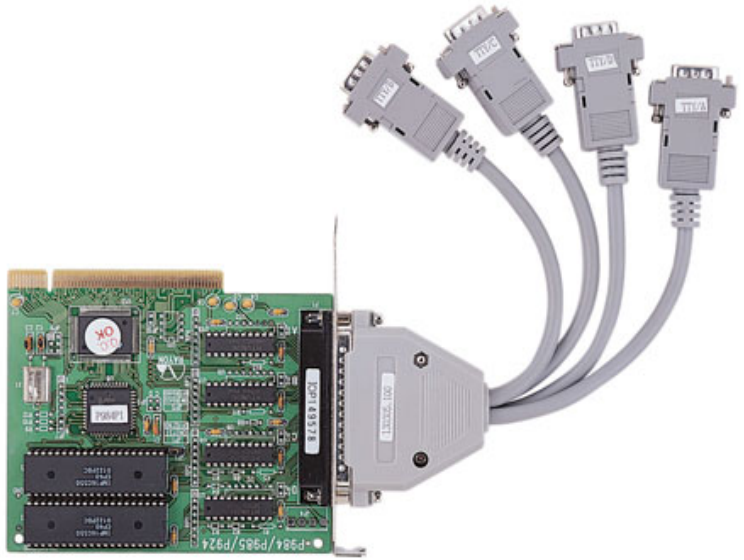
- Network enable hardware.
  - Most HAM equipment is RS-232.
  - Power over Ethernet (POE)
- End-to-end IP to satellite
  - See OMNI project at NASA-GSFC

How do we support legacy systems?



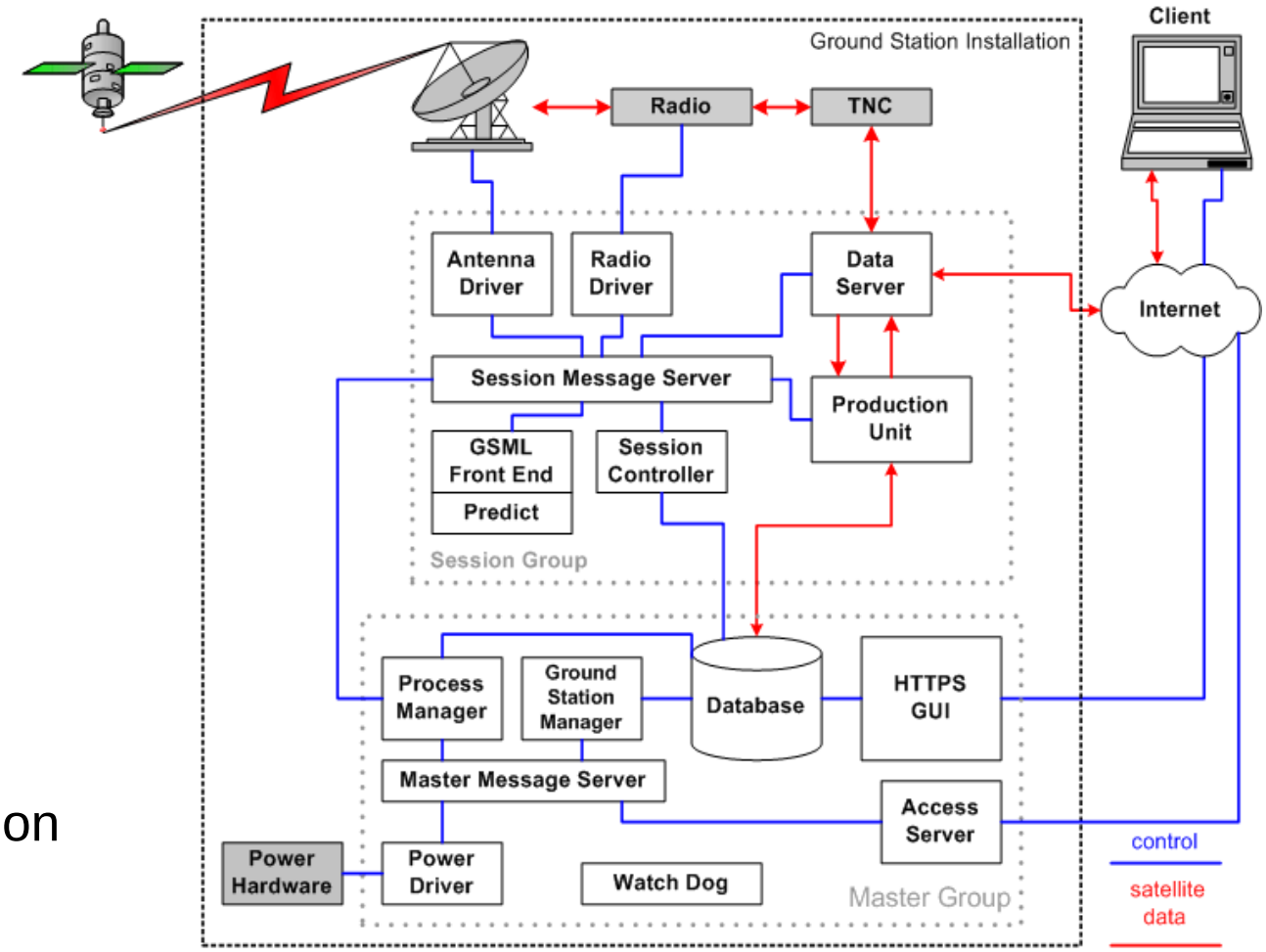
Serial To Ethernet Bridges

- Convert serial ports to ethernet (TCP/IP)
  - Dedicated hardware (Moxa, etc)
  - Hardware and software (Linux + serial port cards)
  - Virtual serial ports, termnetd
- Works for legacy HW and SW

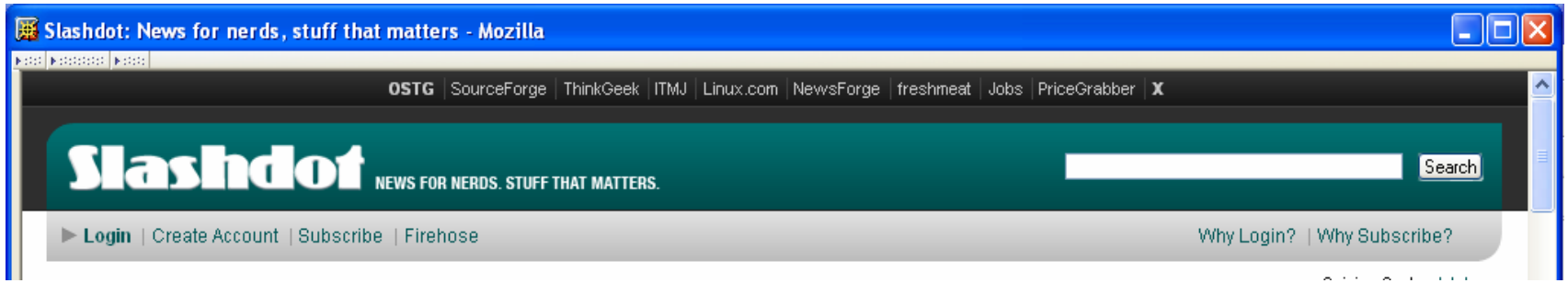


Mercury Ground Software

- Manual or autonomous tracking
- Network accessible (browser and “telnet’)
- Open source
- COTS parts
  - Linux
  - Apache
  - MySQL
- Runs on my laptop
- GSML – ground station markup language
  - Abstract control of station
- Operational, and under development



Lesson: Hackers Lurk Everywhere



- Security is a concern
  - Networked stations are exposed
- Slashdot effect
- Daily/hourly attacks on our machines
- Be prepared and nimble
  - It‘s not as bad as it sounds if you careful
  - We use virtual machines (VmWare, Xen)

Lesson: Failures are a fact!

- Failure not a problem, but a fact!
- We cope with them through ***recovery/repair.***
- Improving recovery/repair improves availability
  - Availability = MTTF / (MTTF + MTTR)
  - Make MTTF very large; then Availability => 1, but, what if MTTR << MTTF
- Failure examples
  - Recent DNEPR2 support
  - Percussive adjustment of radios
  - Fairbanks Internet links

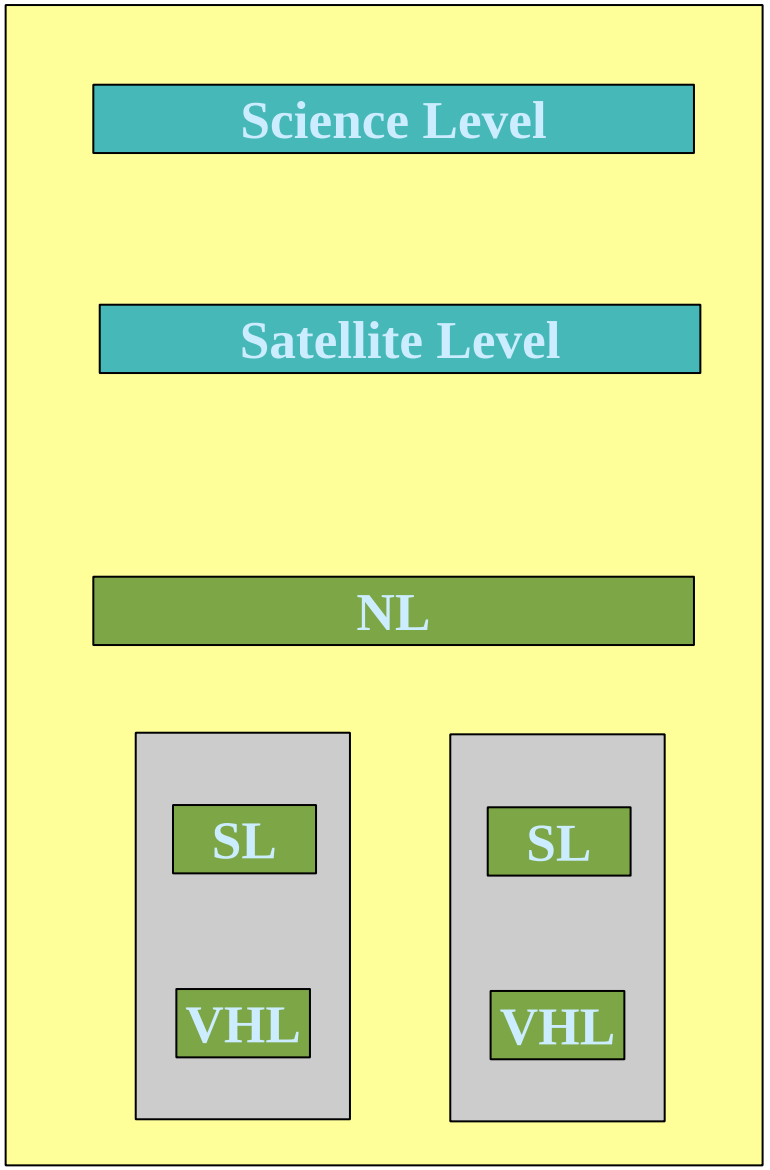
Snow on antennas

Lesson: End-to-End Principle

- "A lower layer of a system should support the widest possible variety of services and functions, so as to permit unanticipated applications" [Saltzer, Reed, Clark].

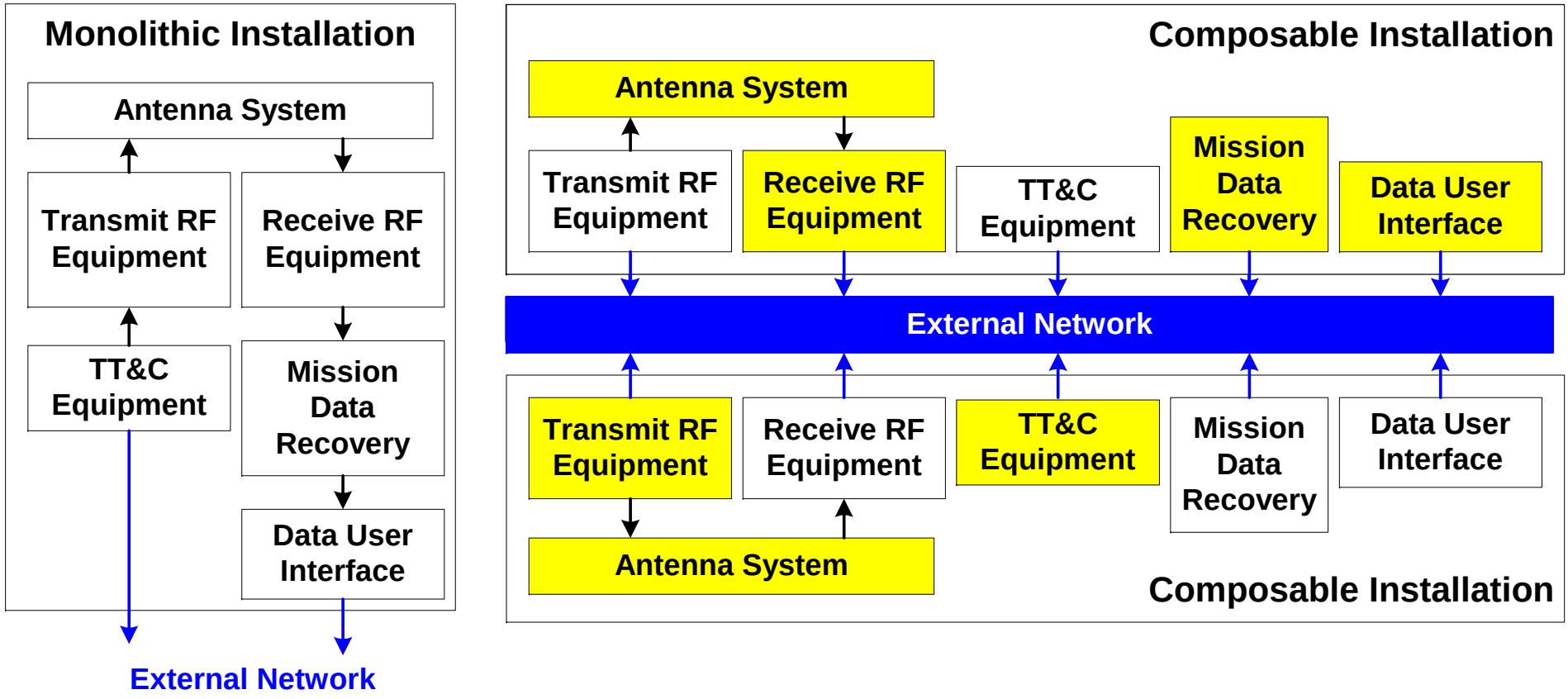
- Virtual Hardware Level (VHL) — fundamental capabilities of low-level hardware.
- Session Level (SL) — typical automation tasks of a single station.
- Network Level (NL) — services of a network of ground stations.

- A *virtual ground station* (VGS)



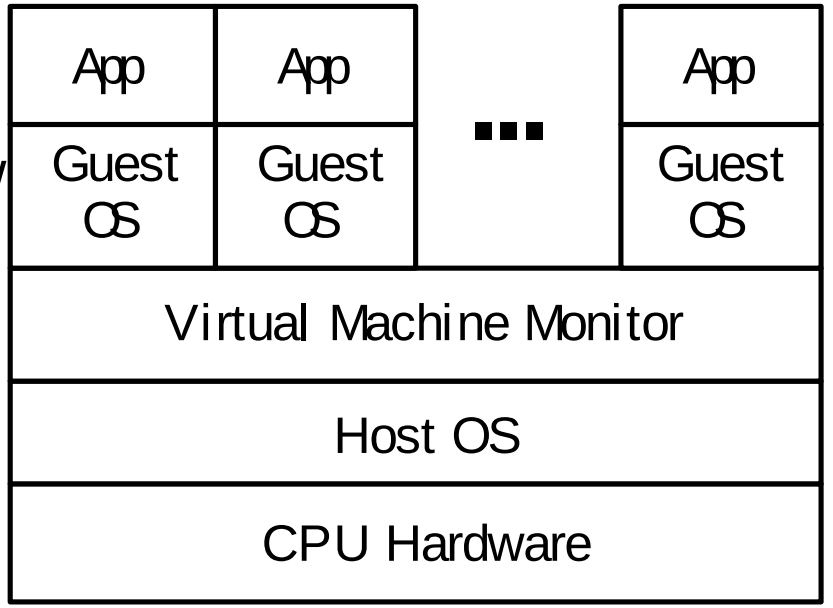
Composable GS

- Distributed GS components can be *composed* to form a *virtual ground station*.
  - A GS is decomposed into core components.
  - These are then assembled to form virtual ground station services.
  - Local teams for optimization, global teams for increased contacts.

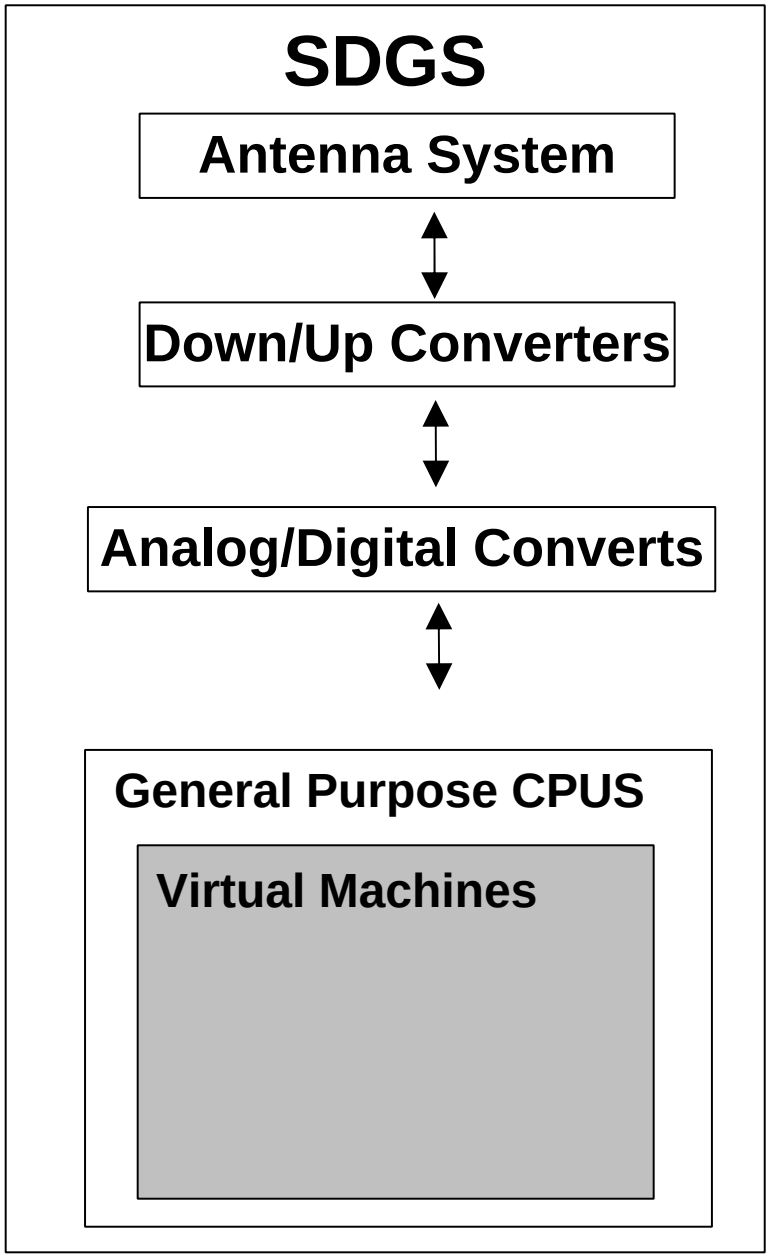


Virtual Machines—A quick digression

- A host OS and hardware running multiple guest OS, the virtual machines.
  - To the guest, it appears to them as if they are the sole machine.
  - To the host, it just appears as a running process. Common place in IBM main frames for years, but now making their way into mainstream computing (ie Vmware, Xen).
- Uses of VMs
  - Guest OS free from the hardware it is running on. Consider HW upgrades now Just copy.
  - Facilitates backups and restorations.
  - Higher utilization of CPU resources.
  - Isolation, sandboxing, and security.
- What if a core GS service included the ability to run a VM?



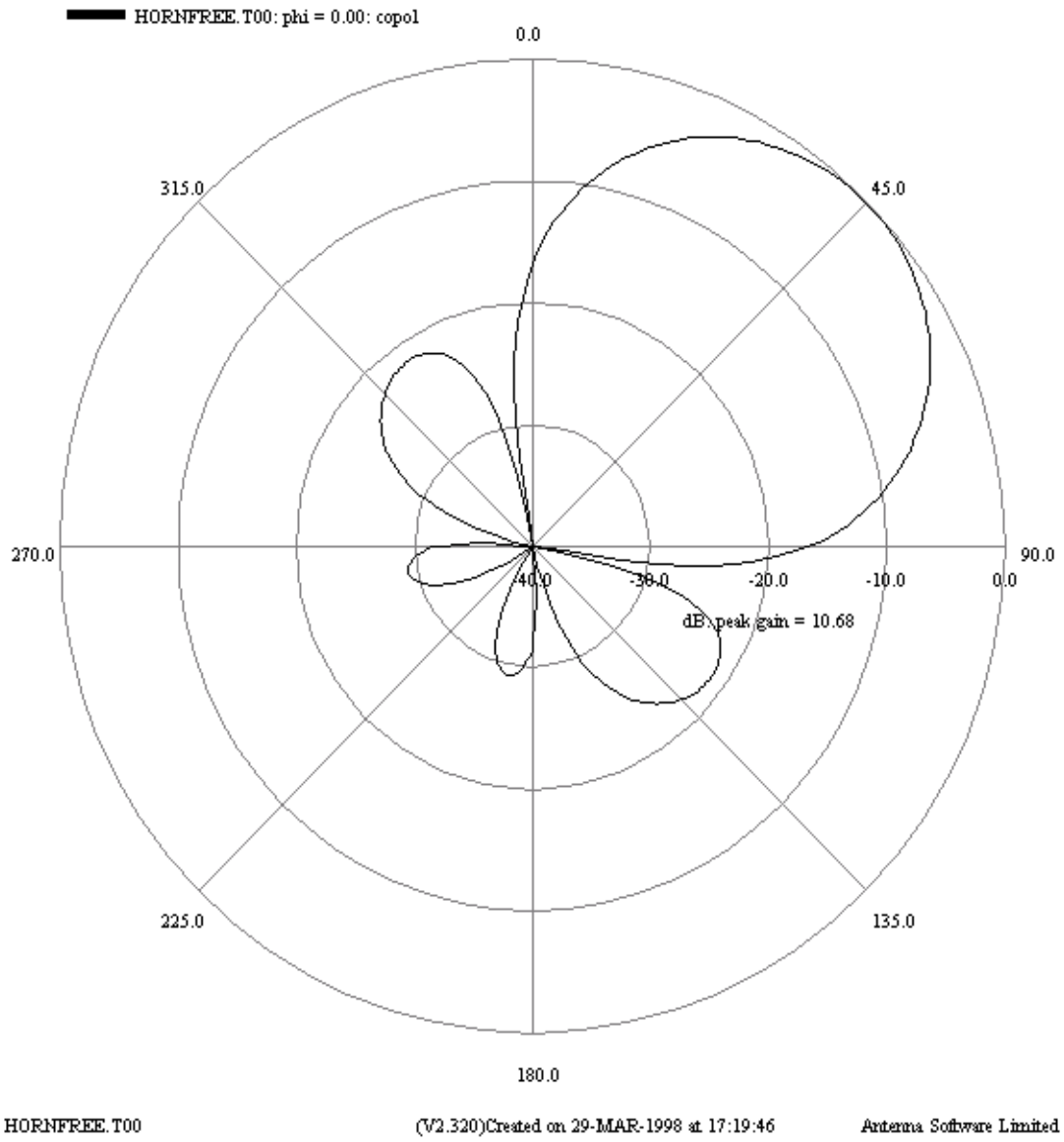
Software Defined Ground Stations



- Reduction in custom hardware
  - Antennas, amplifiers, up/down converters, ADCs, DACs
- Move everything else into a VM
  - Bit sync, FEC, packetization.
  - TTC, misson data, etc.
- VMs are now:
  - Portable
  - Upgradable
  - Customizable
  - Etc.
- Check out gnuradio.org

Lesson: Beamwidth!

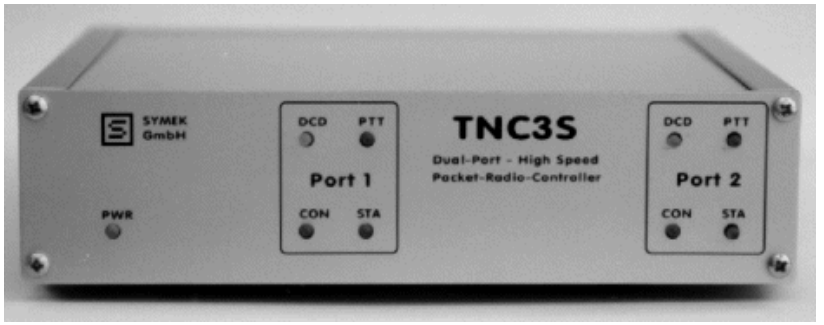
- Sweet spot for tracking
  - Our experience: 20-60 degrees
- Lead the spacecraft, don't follow
- How do we hunt for the satellites?



http://www.maasdesign.co.uk/maas/qmwleo/img011.GIF

Lesson: Make State Available

- What state is your system in?
- View the GS as a *satellite*
  - Telemetry
  - Health management
  - Beacons
- Don't assume what operators will want!
- Make everything as simple as possible, but not simpler - Albert Einstein



Community Efforts

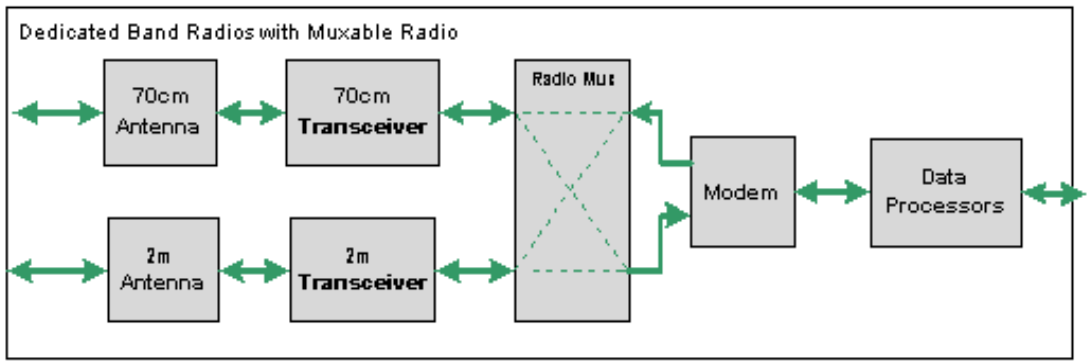
- Non-pointing antennas?

- A strategy for beacons—post launch, pre object ID
  - Power/beacon frequency trade-offs
  - CW beacons have been great!
    - Can we build a low cost receiving station (Niederstrasser)?

- A strategy for object identification
  - CalPoly's IRC is great!
  - Spread spectrum radio with a ground software-defined radio.
  - Beacon packet relays ala ANDE/RAFT

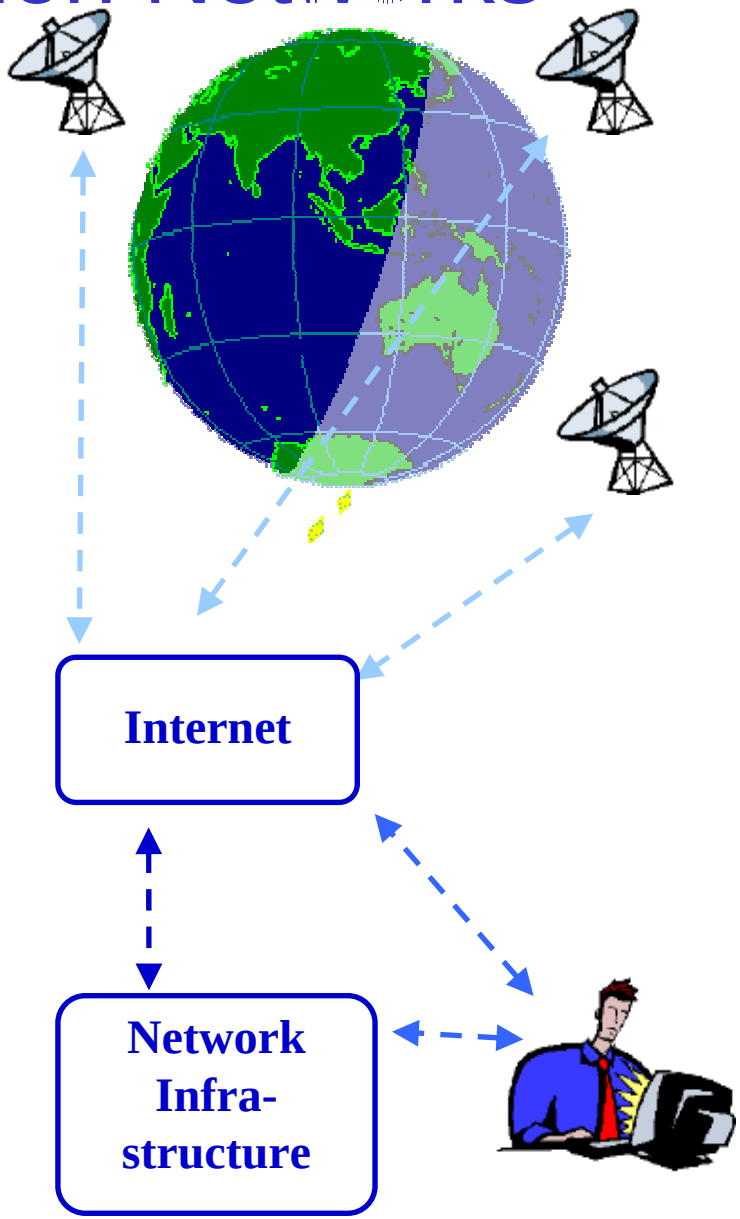
- Matrix switch between radios/modems

- Mass deployment of $15k stations?



Next Steps – Ground Station Networks

- A federated ground station network (FGN)
  - 100's of stations under different administrative domains—universities.
  - Globally distributed facilities that can dynamically join and leave the federation.
  - Heterogeneous and networked via Internet.
- Ability to designate teams of stations
  - Teams collaborate on high level task (e.g. "track this spacecraft").
  - Global teams to increase access windows.
  - Local clustering to optimize ground stations and provide path and node redundancy.
- **Prototype Mercury system**
- **Global Educational Network for Satellite Operations (GENSO)**



The End