Planetary LakeLander - A Robotic Sentinel to Monitor Remote Lakes

Liam Pedersen
Carnegie Mellon University
NASA Ames Research Center
Moffett Field, CA, USA
`Liam.Pedersen@nasa.gov`

Trey Smith
Carnegie Mellon University
NASA Ames Research Center
Moffett Field, CA, USA
`Trey.Smith@nasa.gov`

Susan Y. Lee
Stinger Ghaffarian Technologies
NASA Ames Research Center
Moffett Field, CA, USA
`Susan.Y.Lee@nasa.gov`

Nathalie Cabrol
Carl Sagan Center, SETI Institute
NASA Ames Research Center, Space Science Division
Mountain View, California, USA
`Nathalie.A.Cabrol@nasa.gov`

Abstract

This field report describes the design and operations of the Planetary LakeLander (PLL) probe and its ground data systems. LakeLander's primary mission is to characterize the physical, chemical and biological processes occurring in a high-altitude lake, and how they are being impacted by rapid deglaciation. LakeLander's secondary purpose is to test operations concepts for future exploration of Titan's lakes.

The LakeLander probe is a permanently anchored buoy that measures both surface meteorology and water quality parameters in the top 40 meters of the water column. The concept of operations calls for the probe to continue collecting and downlinking data through the Andean winter, when the lake is inaccessible; this drives the power system design and forces a strong focus on system reliability, analogous to a space mission.

The PLL ground data system provides the central archive of downlinked data. They are structured around a unified data sharing web site that includes tools for mapping, data visualization, documentation, and numerical analysis. The web site provides a hub for

engaging the science team and enables interdisciplinary collaboration.

This report concludes with lessons learned during field deployment and several months of remote operations on the lake. Among the conclusions: (1) The choice to use an off-the-shelf profiling system has proven wise; (2) effective maintenance of a long-lived remote system requires extensive measurement, logging, and display of as many system variables as possible; (3) and the visualization sandbox component of the data sharing web site has made numerical analysis of probe data much easier and more accessible to the entire interdisciplinary science team.

1 Introduction

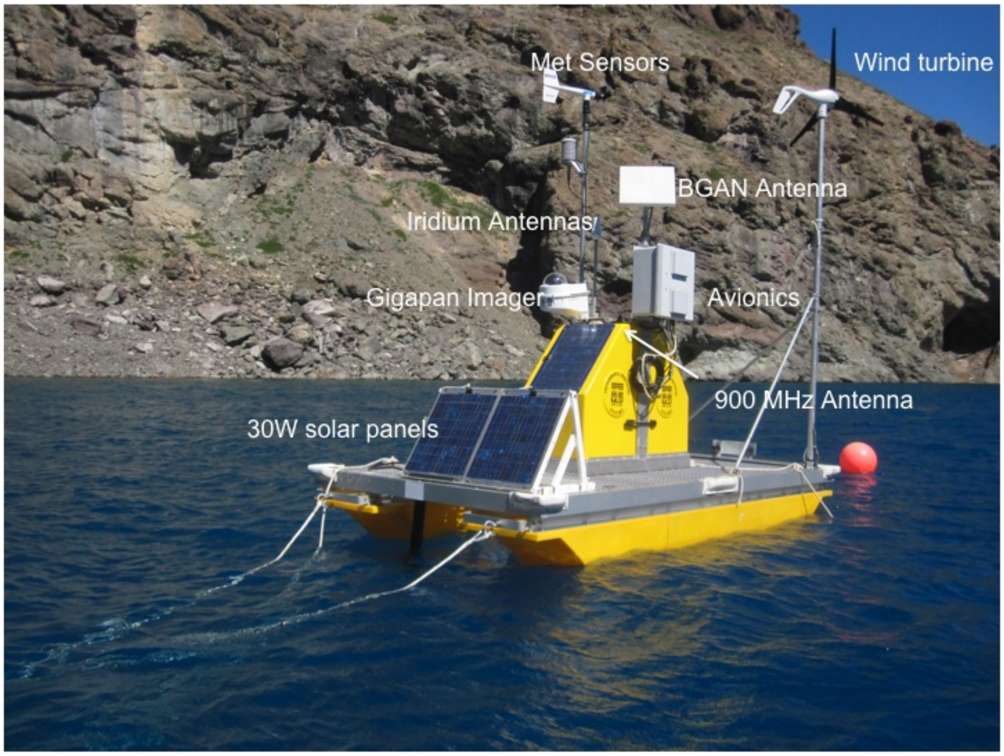


Figure 1: LakeLander probe at its permanent mooring location in Laguna Negra. LakeLander features meteorological sensors, a commercially available water profiling system, cameras, computation, communications, power and thermal subsystems.

One of the key goals in environmental science is to measure baselines, detect changes, and correlate these to driving factors. The LakeLander robot (Figure 1) is designed for the task of monitoring remote lakes, and as an operational test bed for the proposed Titan Mare Explorer (TiME) mission to land in and explore the methane lakes of Titan (ESA/NASA, 2009; Stofan et al., 2009). Titan is the only body in the solar system

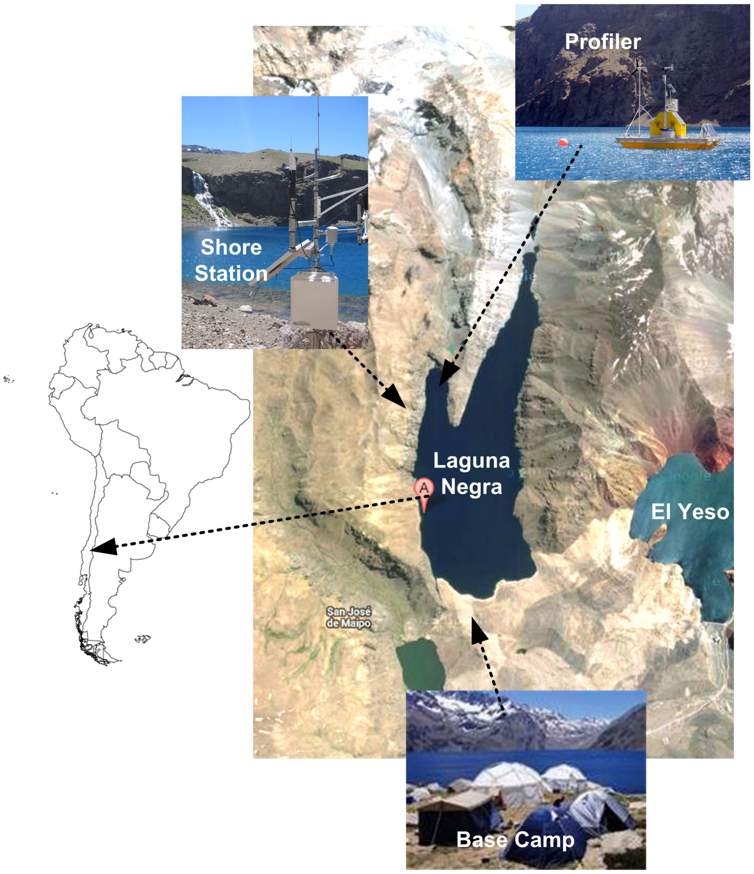


Figure 2: Laguna Negra (33°39′S/70°07′W) is a 6 km × 2 km, 300 m deep glacial lake near Santiago, Chile. To the north of the lake is the rapidly retreating Echaurren glacier. Deglaciation is associated with changes in meltwater discharge into the lake, with undocumented impacts on the ecosystem and biodiversity. This region is one of the main freshwater resources for the capital region of Santiago. Background credit: Google Earth.

besides Earth known to have lakes and rivers.

LakeLander is permanently anchored in Laguna Negra a high altitude alpine lake in central Chile (Figure 2), and has monitored the lake continuously since December 2012 to the present.

LakeLander's principal goals are the collection and transmission of environmental measurements, and to serve as a platform for researching onboard science data analysis deemed necessary for the power and communications limited TiME mission.

This paper is a field report of LakeLander's first months in operation. It focuses on LakeLander's hardware and software design space, and the ground systems for archiving and distributing the data. The scientific results and data analysis algorithms are published elsewhere. The design sacrifices state-of-art performance in order to use well-tested off-the-shelf components and meet budgetary and schedule constraints. Key design drivers are explained throughout.

2 Scientific Mission

LakeLander is part of a multi-disciplinary study of the effect of rapid deglaciation and other environmental disruptions on the lake physics, chemistry and biology (Cabrol et al., 2012). Its primary mission is to collect baseline meteorological and limnological data (Table 1) from the surface to 45m depth in order to characterize lake processes and their interaction with surface conditions.

This project traces back to the High Lakes Project (HLP) which explored early Mars lake environment analogs (Cabrol et al., 2009; Cabrol and Grin, 2010), and showed the impact of climate variability on the geophysical environment of several lakes located in the arid Andes (Bolivian and Chilean Altiplano and Andes 18-23.5°S). These lakes were formed by deglaciation at the end of the Pleistocene, but glaciers have disappeared since the onset of aridity 10,000 years ago. They are now barely sustained by limited and variable snow precipitation (30-90 mm/year) and experience strong negative water balance (-1,200 mm/yr). HLP demonstrated the domino effect of precipitation variability and intra-seasonal shifts in temperature, cloud fraction and wind regime. This change in the amount and nature of aerosol triggers a chain reaction in the lakes, changing water chemistry and transparency (Cabrol et al., 2009).

LakeLander is a test bed for the autonomous analysis of limnological and meteorological data. This is desired because lake conditions can change rapidly, at times requiring a high rate of sampling that cannot be sustained by available power. Lake changes are driven by such factors as glacial discharge, storms, seasonal state changes (e.g. stratification), and unexpected stochastic events (e.g. landslides, hydrothermal discharge). Detecting and responding to these in a timely manner is expected to improve data collection and alert the science team when something interesting occurs and additional resources (e.g. satellite data) need to be brought to bear.

3 Related Work

Dunbabin and Marques (2012) surveyed 20 years of robotics for environmental monitoring. A majority of applications are marine, utilizing either Autonomous Underwater Vehicles (AUVs) or Autonomous Surface Vehicles (ASVs) for mapping and survey tasks. They identify the following challenges for terrestrial environmental monitoring robots: (1) reliability, safety and endurance, (2) human-robot interfaces, (3) adaptive mission planning, (4) real time dynamic process tracking and (5) event detection and classification. We address (1) and (2) in this paper, and will be addressing (3)-(5) in the subsequent research for which this platform is intended.

Hitz et al. (2012) described the development of an ASV for both horizontal and vertical lake profiling, equipped with the same limnology sensor payload as LakeLander.

Long term, high frequency observations are essential for characterizing ecosystems, particularly around threshold transition events that should exhibit increased variability immediately prior to their occurrence (Carpenter and Brock, 2006). Along these lines, the Lake Lander observed rapid lake changes during seasonal transitions (lake fall and spring turnover), snow melt run-off during periods of higher temperatures, and even occasional landslides.

4 Hardware

LakeLander is built around a commercially available solar powered profiling system (YSI, 2012) that features a winch to raise and lower a water quality sonde through the water column. Added to this are additional sensors, computation, communications, thermal and power subsystems.

An additional (solar powered) meteorology station is positioned on a nearby shore (Figure 3) to collect additional meteorological data and measure micro-scale meteorological differences. The common sensors between the shore station and the LakeLander are identical. The shore station is managed by LakeLander using a 900MHz radio link.

4.1 Scientific Payload

LakeLander's primary payload is a commercially available water profiling system (YSI, 2012). The profiling system consists of a YSI 6600 multi-parameter water quality sonde that is raised and lowered through the water column by a winch assembly. This unit was chosen because it was the only commercially available profiling system known to us at the time. The 50m long cable permits profiling down to the bottom, which is $\sim$45m at the chosen location. A depth sounder (not used for science) is used to measure depth prior to spooling out the cable in order to avoid damaging the sonde on the bottom.

Table 1 lists the specific water quality and meteorological sensors on LakeLander and the nearby shore

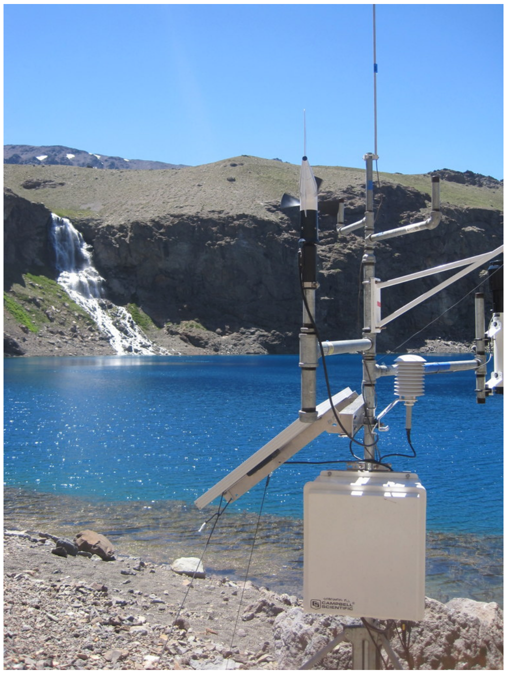


Figure 3: Backup meteorology station on the shore near the LakeLander. This station includes additional sensors for measuring precipitation, solar insolation, soil temperature and lake level.

station. The water quality and meteorological sensors are industry standard sensors requested by the science team and recommended to us by the profiler manufacturer.

The Sanyo Pan-Tilt-Zoom (PTZ) camera system is used to acquire images that cover a 360° field of view around and above the probe. Images are stitched onboard and a thumbnail of the stitched panorama is uplinked daily; the full-size images are stored onboard and can be retrieved occasionally during maintenance visits. Panorama thumbnails are used to gauge cloud cover, surface conditions, input from the nearby waterfall, and visible physical damage to the platform. The PTZ camera is located within a watertight dome enclosure that blocks visibility at downward-pointing angles. In order to ensure visibility of the water surface and shoreline near the cascade inflow into the lake, the enclosure is tilted down about 15° in that direction.

The shore station includes a lake level sensor (a pressure sensor) securely placed 10m underwater in the nearby lake bed. Use of a GPS onboard the LakeLander to measure lake level was considered and rejected because it would not provide the required resolution (1cm) and sampling rate (1 minute intervals).

Table 1: LakeLander Scientific Payload: Description and part number provided when applicable.

Sensor	Properties [Part number]
YSI 6600 V2 Multi-Parameter Sonde Plus Depth Sounder	Sonde depth, pH [6561], oxidation/reduction potential [6150AF], dissolved oxygen [6150AF], turbidity[6136AF], chlorophyll [6025AF], blue/green algae [6431AF], conductivity (salinity)/ temperature probe kit [6560], lake depth [6960]
Onboard Meteorology Station	Wind speed/direction [200067], air temperature [HMP45C-L9], barometric pressure [CS105], relative humidity [HMP45C-L9], magnetic heading [201901]
Sanyo PTZ camera	Image panoramas for monitoring surface conditions (waves, color), terrain changes (landslides, vegetation, snow accumulation), cloud cover, and nearby stream inputs [VCC-HD5400]
Off-board Meteorology Station	Wind speed/direction [05106], air temperature and humidity [HMP45C-L9], barometric pressure [CS105], precipitation [50203], soil temperature (thermistor) probe [10K3MCD1], solar insolation [400612240], lake level (fixed submerged pressure sensor) [600-10]

4.2 Avionics

The original commercial off-the-shelf (COTS) profiling system from YSI uses Campbell Scientific CR1000 data loggers for data acquisition and control. They tolerate cold temperatures (-20C) and consume little power (nominally 12-200 mW when active), but lack sufficient computation and storage for PLL onboard data analysis requirements.

LakeLander's avionics design (Figure 4; Table 2) retains the Campbell Scientific data loggers, but has a FitPC for computation and to control the new subsystems. The FitPC runs Linux, has a 512MB solid state drive (SSD), two network ports and an RS232 serial port. It runs the system executive, stores and analyzes acquired science data, and manages system power and communications to the US. The FitPC was chosen due to its extremely low power consumption (1-8 W).

The design philosophy is to minimize any changes to the well-tested vertical profiler system and to preserve the manufacturer-supported methods of connecting with it, even in the event of other hardware failing. Consequently, the profiler hardware was retained (Iridium modem and serial connections) and could be used as originally designed in the event of catastrophic failure of the avionics subsystem.

Using data loggers allows continuous data acquisition even when the FitPC and other avionics components are inoperative or shutdown to conserve power (12 h/day). The FitPC pulls data off the data loggers at regular intervals for analysis, storage and transmission to the USA.

The FitPC and peripheral units are linked together by an Ethernet LAN, simplifying inter-device communications and making it possible to access each device from multiple points, including base camp (via the 900MHz radio Ethernet bridge), albeit at the cost of increased power use compared to purely serial based connections. The shore-station sensors are likewise logged by a CR1000 data logger, connected to the LakeLander LAN via an additional 900MHz radio.

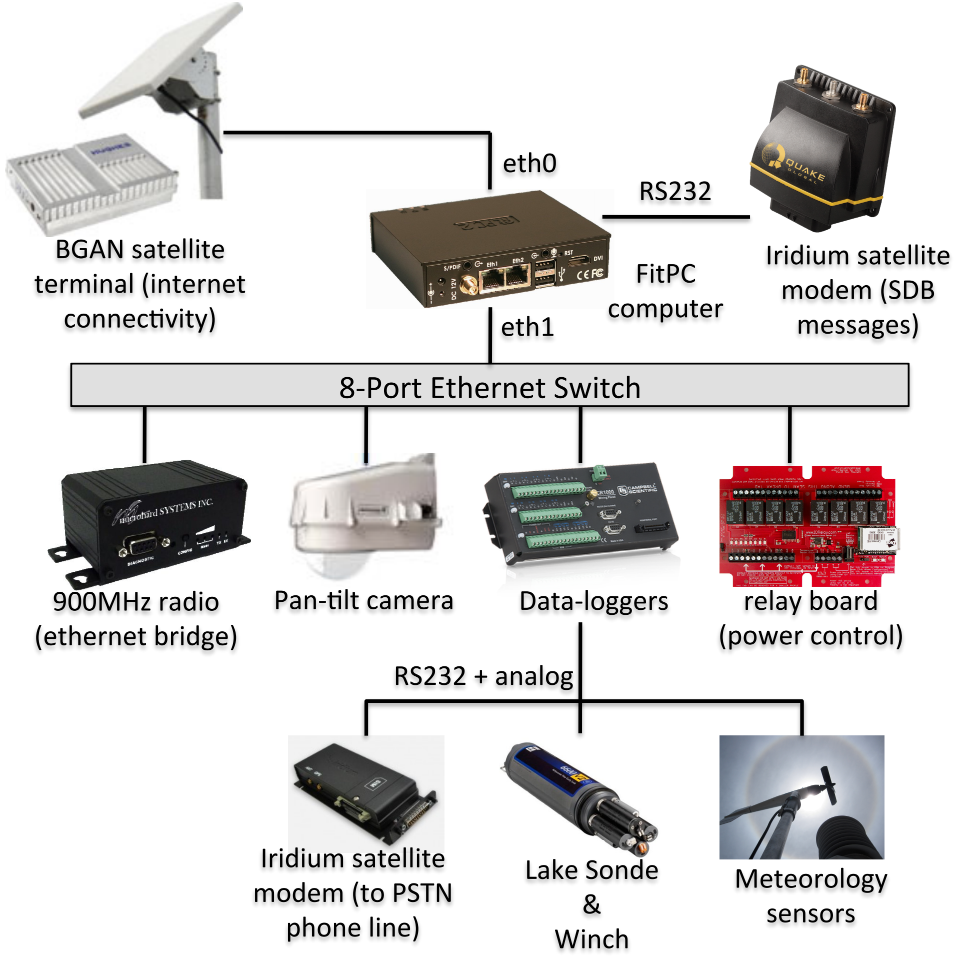


Figure 4: LakeLander's major avionics components and their internal data bus connectivity

4.3 Communications

Reliable communications are critical to LakeLander for data retrieval, system monitoring, commanding and software upgrades. Specifically, ensuring that the system is available for remote login (via the `ssh` secure shell terminal) makes tolerable the risks associated with a compressed development and testing schedule. Consequently the design philosophy is to maintain redundant communication channels (Figure 5, Tables 3-4).

The primary communication channel is via a Hughes 9502 BGAN M2M satellite terminal that provides a globally routable Internet connection through the geosynchronous INMARSAT service. The FitPC is

Table 2: LakeLander Avionics Hardware

Component	Manufacturer and Model	URL
Computer	FitPC with 512GB SSD	www.fitpc.com
8 Port Ethernet Switch	Netgear, GS108	www.netgear.com
Ethernet Power control board	NCD Reactor Board ZETH ME	www.controlanything.com
BGAN terminal	Hughes 9502 BGAN M2M	www.hughes.com
Iridium SBD transceiver	Quake Global Q-Pro	www.quakeglobal.com
Iridium modem	Iridium 9522B Transceiver	www.iridium.com
900MHz radio Ethernet bridge	Microhard IPn920	www.microhard.com
Wind turbine	Southwest Windpower AIR Breeze	www.windenergy.com
Added Solar Panels	Solarex MSX30L	www.getoffthegrid.com/solarex.html
GigaPan weather enclosure	Dotworkz, D2-HB-MVP	www.dotworkz.com

configured as a router for external access to all devices on the local area network. Using the FitPC as a router provides additional power savings by removing the need for a separate router. Microhard Nano IPn920 900MHZ radios extend the LAN to the shore station and to the (summer-time) base camp. A Quake Global Q-Pro modem provides GPS location and a backup communication channel over the Iridium constellation's Short Burst Data (SBD) service. The additional Iridium 9522 modem connected to the data loggers allows telephone dialup access to them.

The 9502 M2M BGAN transceiver uses a 1' × 1' square antenna that must be pointed at a geostationary INMARSAT satellite that is continuously in view. Through trial and error an approximate beam width of 30° was established, sufficient to encompass the measured 15° of platform yaw range (the platform is tightly moored at both ends to prevent yawing and drifting around a single anchor point). The 9502 was chosen because of its reduced power consumption, separated antenna, and low re-connection costs. LakeLander is programmed to periodically power up the 9502 and maintain a connection, during which time data is transferred (via the `rsync` file transfer protocol) and an operator may log into the system (via the `ssh` secure shell terminal). Port forwarding on the FitPC allows the operator to work directly on other hardware attached to the LAN via configuration pages and data logger ports.

The two Iridium transceivers use omni-directional antennas to communicate with the Iridium satellite constellation in Low Earth Orbit (LEO). Connectivity is only possible when the satellites are overhead. Most of the time, they are close to the horizon and likely to be occluded by the surrounding high mountains at the mooring site (significantly better conditions existed at the base camp 2 km away). Connection times of 5 to 10 minutes were observed at the field site.

The Q-Pro SBD modem is intended to send brief command strings to, and receive short status reports from

LakeLander independently of the BGAN satellite system. The Iridium SBD service allows a transceiver to send and receive small text messages, that are relayed as email attachments to the ground segment. The messages are queued in the system until the satellites are in view (and the transceiver is powered up), and typically get delivered within minutes.

The remaining (legacy) communications channel is an Iridium 9522B modem, with a serial connection to the profiler data loggers. A *dialup* connection is established from the ground segment, with rates between 1200 and 2400 bps.

Local communications between LakeLander, summer base camp, and the shore station are established by 900MHz spread spectrum Microhard IPn920 radios, configured as Ethernet bridges in a one to many network topology, with LakeLander as the master unit. This extends the LakeLander LAN to include the base camp LAN and the shore station, with the proviso that no direct communication between base camp and the shore station is possible.

Table 3: LakeLander available communication channels

Channel	Medium	Bandwidth	Latency	Cost
BGAN	IP	<1 Mbps	<2s	$5 / MB
Iridium SBD	Text message	340 bytes/msg TX 270 bytes/msg RX	minutes	$34/month up to 30 KB
Iridium CSD	RS232	1400-2400 bps	n/a	$1.5/minute + long distance charges
Microhard IPn920	IP	<400kB/s	60 ms	n/a

Table 4: Data upload budget

Data	*Nominal Rate*	*KB / day*	*MB / month*
Water profiles	24 / day	192.9	5.8
GigaPan panorama thumbnail	Daily	127.5	3.8
Meteorology data (onboard)	15 min	19.2	0.6
Shore station data	15 min	23.0	0.7
Engineering data		1.4	0.04
Total		**363.9**	**10.9**

4.4 Power System

Constrained power is the biggest hurdle to operating reliably without interruption for an entire year. The original profiler power system (two 30W panels facing North and South respectively) proved sufficient only to operate the COTS profiler several times per day during the 2011 austral summer months when LakeLander is exposed to less than eight hours of direct sunlight on cloudless days. A crude estimate using ephemeris data

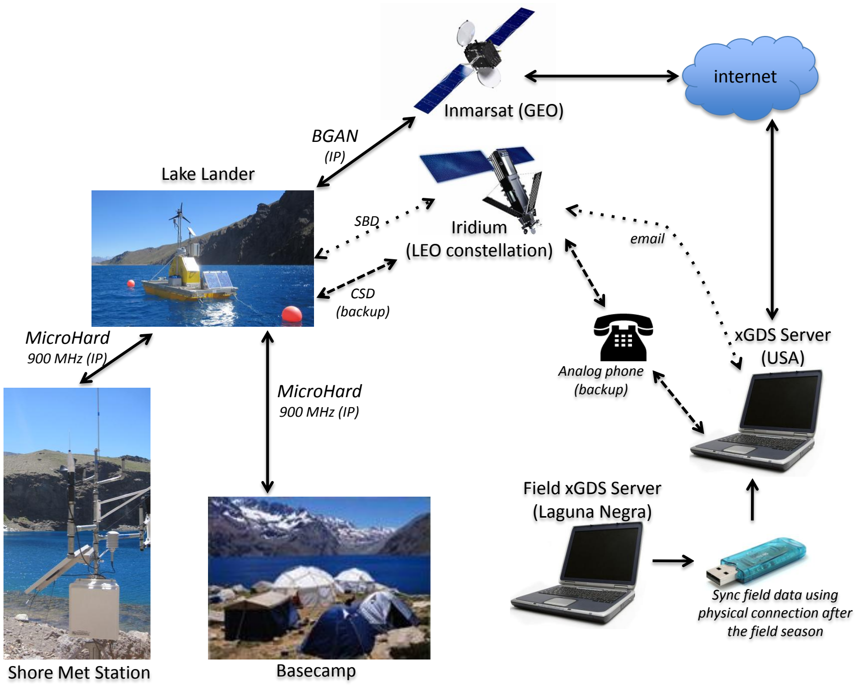


Figure 5: LakeLander communications and data flow

and Google Earth suggests the probe will get at most four hours of direct sunlight during the winter solstice. Powering the full system and increasing profiling rates to 24 profiles per day requires both additional power sources and demand management.

Table 5 lists system power requirements. Profiling 24 times per day and running all systems continuously would require roughly 430 W h/day, well in excess of the estimated 150 W h/day available from the original 2 solar panels. Two additional 30W north facing solar panels and a Southwest Windpower AIR Breeze marine wind turbine increase expected energy availability to 450 and 270 W h/day in summer and winter respectively (Table 6). Unfortunately, weather conditions at Laguna Negra are highly variable and exhibit significant small scale geographic variation. Weather data for the probe location for a full year is not (yet) available, which limits confidence in these power generation projections.

Power consumption is managed by turning off selected subsystems when they are not needed, and by varying the number of profiles acquired per day according to a set of operating modes: *standard*, *conserve*, and *survival* (Table 5). Standard and conserve modes are designed to allow maximal sampling rates in summer and winter respectively. Power consumption is dominated by the winch, which operates for approximately 15 minutes per profile, or 6 hours per day during standard operations (24 profiles/day). To maximize the power available to the winch (and the CR1000 that controls the winch), the probe turns off the FitPC and communications gear from 12 to 23 hours each day. The daily rebooting of the FitPC has the added benefit of restoring the system to a known (operating) state each morning.

If the battery voltage drops below a threshold, the FitPC and communications gear is shutdown until the voltage rises above a secondary (higher) threshold, at which point the power management board turns everything back on. Although the data loggers and profiler can continue to operate independently without the FitPC, no communications or data processing is possible. Survival mode is designed to help avoid this emergency situation for which there is no guaranteed recovery time.

Table 5: Estimated LakeLander daily power requirements for different operating modes. Estimates for off-the-shelf components and winch motor are based on nominal manufacturer specifications. These values were also compared with logs of average power consumption for components controlled by the power control board. Winch power estimates are based on estimated daily duration of winch motion (number of profiles/day as set by the executive, multiplied by 15 minutes of active winch motor power per profile).

Component	Power [W]	Standard Mode Hours / day	Conserve Mode Hours / day	Survival Mode Hours / day
FitPC	6.0	12	6	1
Ethernet Switch	2.5	12	6	1
Quake Global Comm	0.1	12	6	1
BGAN	8	12	6	1
GigaPan Voyage	24	0.3	0.3	0.3
Power Controller Board	1.2	24	24	24
Microhard Radio	0.4	24	24	24
CR1000 electronics	2.4	24	24	24
Winch	20	6	3.3	1
Total	**65**	**432 W h/day**	**268 W h/day**	**142 W h/day**

Table 6: Estimated LakeLander daily power generation for summer and winter. The estimated solar panel power generation times are estimated from the duration of direct sunlight less 25% to account for cosine losses. Cloud cover further reduces the total energy budget. The wind turbine power generation is based on the observed average wind speed and manufacturer provided power conversion chart.

Component	Power [W]	Summer Hours / day	Winter Hours / day
Built In Solar Panels	30.0	5	2
Additional Solar Panels	30.0	5	2
Wind turbine	7.5	20	20
Total	**68**	**450**	**270**

4.5 Thermal System

The thermal system design is based on initial reports that in a typical year the surface air temperature at Laguna Negra ranges from -10 to 25° C. The probe has since measured a similar temperature range. All components of the probe are designed to operate over this range except the FitPC solid state drive, which is only rated down to 0° C and therefore presents a thermal management challenge.

LakeLander uses a combination of insulation, fans, phase change materials, and system timing to keep the FitPC in the 0° to 60° C temperature range. The avionics box is sufficiently well insulated that during summer there is a concern that the FitPC may overheat. Fans are activated to draw in outside air when CPU temperature exceeds a threshold. In addition, phase change materials (a plastic bag of paraffin wax) are in contact with the FitPC. If its external temperature rises above 50° C the wax slowly melts, absorbing significant heat as it does so. This is designed to allow the system to ride out brief periods of peak afternoon temperatures. The wax re-freezes overnight.

The FitPC should ideally be run at night (when temperatures are lowest) so that its own waste heat keeps it warm. Unfortunately, this conflicts with the power management that requires the biggest electrical load be during the day when sunlight illuminates the solar panels. The chosen compromise was to turn on the FitPC no earlier than 11am (local time) every day at which time the ambient temperatures have risen and the batteries have had time to charge.

5 Onboard Software

LakeLander is a complex system with a variety of integrated hardware and strict power management requirements. It must operate for a year on its own, managing its sensing activities, recovering from faults, and uplinking its data via satellite. In future development LakeLander's adaptive science module must be able to adjust sampling rates or select the most interesting data for uplink based on onboard analysis of science data. These requirements drove the design of the onboard software system (Figure 6).

5.1 Design Choices

The design choices for LakeLander's software were motivated by themes of system reliability, developer productivity, lightweight rapid prototyping, and modular design for reuse. In some cases, tools were selected that come primarily from the web development community and may not be familiar to robotics practitioners. This discussion will hopefully broaden awareness of the trade space.

For our operating system and platform, we chose Linux/x86. That is the primary development environment for our research group and is very suitable for embedded hardware. Relative to power-saving ARM processors,

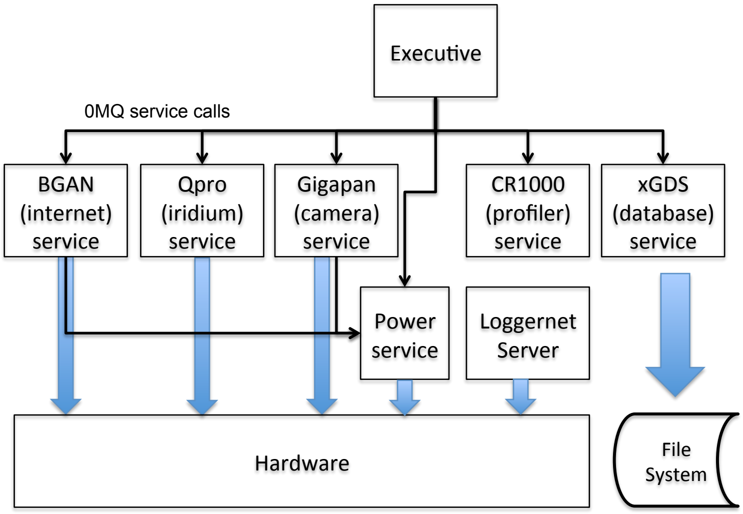


Figure 6: LakeLander's software architecture is a set of Python processes, each providing a specific service and communicating using the 0MQ messaging system. The executive service handles command sequences from the ground segment and scheduled activities (e.g. collecting daily image panoramas). Other services manage access to hardware. Data collection is run on independent (always on) CR1000 data loggers. The CR1000 service handles pulling stored data from the data loggers and checking their status through the proprietary LoggerNet Server software.

choosing an x86 processor helped us reduce development time due to the wider range of available drivers and software packages. The processor performance was also overspecified to allow comfortable headroom later for running onboard adaptive science algorithms.

Our primary programming language for new code was Python. Python is a high-level programming language that is known for its readability, brevity, strong standard library, and active community. However, if used naively for low-level CPU-intensive jobs, it often performs 10-100x slower than compiled languages such as C++, and its use of garbage collection can cause problems in systems with hard real-time constraints. Python was selected because hard real-time performance was not a major concern, the developer team had substantial development experience with Python, and developer productivity was the paramount consideration.

Our overall software system is structured as a "service-oriented architecture", i.e. it is split into several independent services that communicate largely by calling each other's API functions. This modularity facilitates independent testing and should make it easier to reuse a subset of modules on future projects. The design was inspired in part by the service-oriented architecture our research group uses for planetary rovers, written primarily in C++ (Flückiger et al., 2008). Building services on the Robot Operating System (ROS) (Quigley et al., 2009) was considered but rejected because of hardware constraints and administrative requirements at our workplace that prevented running a compatible Linux distribution.

We chose to implement our services as separate UNIX processes. Relative to using multiple threads in a single process, this approach better isolated services. In our design it was important to isolate high-reliability core services, such as the exec, from problems that might occur in less critical services. Multi-process was also a good fit for the Python environment, where multi-threading support is relatively weak; all Python threads must compete for a "global interpreter lock" to complete certain operations. The multi-process approach required communicating via message passing rather than shared memory, which added overhead, but the impact was minimal for our application.

When concurrency was needed within a single service, we used multiple cooperative green threads based on the Python gevent library. Green threads impose less overhead than OS threads, and are a good fit when a process spends most of its time waiting for I/O events and does not need to use multiple CPUs. They are also easier to use with external libraries that are not designed to be thread-safe.

For communication between services, we used ZeroRPC, a lightweight and language-agnostic remote procedure call library. It allowed us to implement the services as methods of normal Python classes, then make them available to network peers through a few lines of ZeroRPC wrapper code. For rapid development, ZeroRPC also offers a simple command-line client which you can use to invoke API methods of the services, introspect what arguments they require, and browse documentation. Note that for LakeLander, ZeroRPC runs over TCP on local network links—it may not be suitable for more challenging networks where specific quality of service rules are necessary to get good performance.

The system was designed to support periodic reboots of the main processor. Limited onboard power resources required shutting down the main CPU overnight. The power board was also configured to power cycle the FitPC if it missed periodic heartbeats from the onboard software. These policies enhanced reliability since many possible fault conditions (e.g. services hanging or crashing, kernel panic) can be corrected by a reboot.

Because the lake is inaccessible during the winter, reliability was a critical concern. The onboard software is roughly split into two availability tiers: there are the core services (executive, BGAN, and Q-Pro services), and all others. As long as the hardware, operating system, and core services function at a baseline level, the probe calls home regularly and software updates can be applied to repair any problems that arise. Thus, the key safety consideration for software updates is to ensure that the core services remain functional after the update.

Another observation is that, because the probe has a simple and stable safe mode and much of its data collection is managed by independent data loggers, it can tolerate most functional outages even in the core services without significant data loss, as long as the outage is remedied within a few days.

These considerations led us to develop a *version rotation* scheme for software updates. The probe keeps several version snapshots of the onboard software in its local storage. At daily boot it selects which version to execute based on a rotating schedule. When a new version is deployed, it is added to the rotation. This provides assurance that if the core services fail when the new version is loaded, an old version will eventually rotate back in and operator control will be regained. Once the new version has been thoroughly tested, an operator can push back the date at which old versions rotate back in, or disable the rotation entirely.

5.2 Software Services

This section provides a brief functional overview of the ten services that compose LakeLander's onboard software.

The *Executive* or "exec" coordinates onboard software activities. It has a single command queue so that all commands routed through it are executed serially, avoiding conflicts. Many periodic activities, such as taking profiles or turning components on and off, are triggered by exec timers. When an exec timer fires it adds commands to the queue. The configuration of the timers is regulated by the exec mode; changing modes is the main way LakeLander regulates daily power consumption. Commands can also be added to the exec queue using Q-Pro messages or by running a command-line client while logged in remotely through the BGAN link.

Several of the services act as hardware drivers. The *Power* service controls the power board. It provides an API for turning components on and off, measuring temperature, bus voltage and power consumption, and scheduling sleep duration for the FitPC. The *BGAN* service controls the Inmarsat BGAN satellite communications terminal. It provides an API for starting and stopping BGAN network connections and uplinking data. When the BGAN is connected, the FitPC can accept a remote terminal login. In case of problems with LakeLander, this login capability is the primary means of repairing the system without requiring a site visit. It can also be used to update the onboard software as new capabilities are developed. The *Q-Pro* service controls the Quake Global Q-Pro satellite comms system. It provides an API for sending messages to the ground and checking for new incoming messages. The exec sends a daily Q-Pro status message and also sends messages after important events. Received messages can trigger exec command sequences. On the ground side, users interact with the Q-Pro system by sending and receiving email. The *GigaPan* service controls the GigaPan Voyage / Sanyo PTZ camera system. It provides an API for acquiring a panorama or a time-lapse movie, stitching panorama frames into a single image, and generating thumbnail previews for daily uplink over the bandwidth-limited BGAN connection. The *Profiler* service controls the profiler data logger that manages the winch/sonde water sensor system. It provides an API for fetching data from the profiler to the FitPC, changing the data logger's stored program (for example, changing how often profiles are acquired), checking its status, and resetting it if it gets into a bad state.

Finally, there are four support services. The *Process Manager* starts the other services at boot and ensures that their console output is logged for later debugging. It provides an API for checking the status of services and restarting them if needed. The *Health Monitor* service checks system status for problems and suggests repair actions to the exec. Currently this is limited to checking the average battery voltage over the last 72 hours and recommending an operating mode for the next day based on how much power is available. The *Data Importer* service imports data from the various data loggers onboard LakeLander and shore station into an onboard MySQL database, where they are available for queries by the Health Monitor and Adaptive Science services. The *Adaptive Science* service is not developed yet; it will monitor science data and recommend changes to sampling rates or select high-value data for uplink.

6 Ground Data Systems

The Exploration Ground Data Systems (xGDS) Project in the NASA Ames Intelligent Robotics Group is developing a web-based software platform that handles mission data for science operations (Lee et al., 2012; Deans et al., 2013). The platform includes tools for planning, monitoring, visualization, documentation, analysis, and search. xGDS has supported diverse operations ranging from rovers and astronauts at the Haughton Mars Project and Desert RATS to submersibles and divers at NEEMO and the Pavilion Lake Research Project.

A unified xGDS repository of PLL science data (Table 7) has been established, thanks to strong support from the science team. PLL science themes concern correlations between different data sets (example: turbidity and chlorophyll), so making it easy to visualize and analyze all data through a single interface is a major benefit. The system helps us track what data sets have been collected, ensure proper backups, and establish common meta-data standards to support search (uniform sample numbering, GPS locations, timestamps, attribution, and so on). Users interact with xGDS primarily through a web browser, so the whole data repository can be made available to the international science team with minimal overhead for software installation and maintenance.

Data set	*Characteristics*
Water quality	18 time series, 2 platforms, variable depth, 4 months of data
Weather	16 time series, 2 platforms
Probe health	7 time series
Biology samples	71 samples, 3 teams recording different sample parameters
Geology samples	6 samples
Map layers	13 map layers, several sources
GigaPan imagery	10 gigapans, 15 gigapixels

Table 7: PLL 2012 Data Sets Accessible Through xGDS

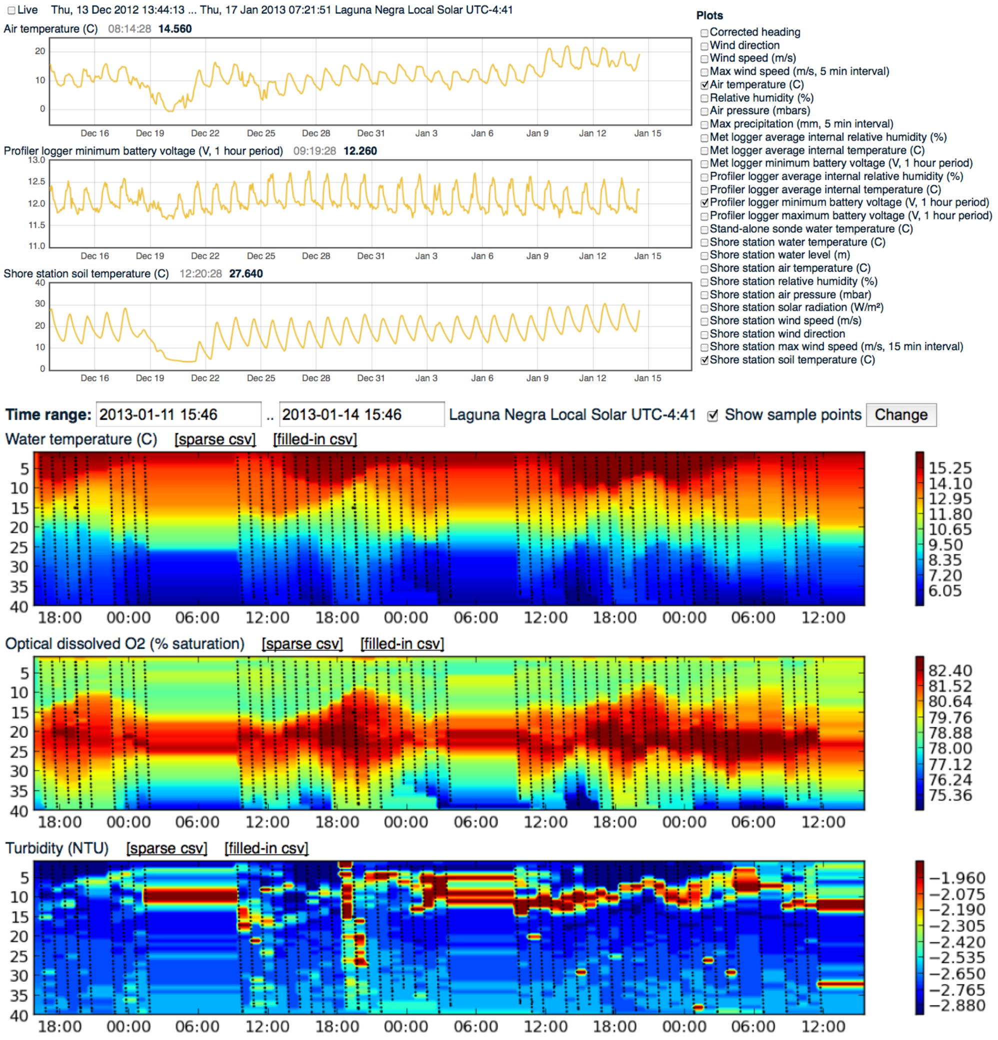
Data flow is a concern given the probe's reliance on limited satellite bandwidth. Figure 5 shows the data flow for 2012 operations. During the field season, data was first collected and annotated at a server in base camp where the field team could access it. LakeLander robot data was transferred over the local Microhard radio link and a variety of other data sets were transferred by portable USB drives or manual data entry.

After the field season, the field server was carried back and its data was synced to a fixed server where the science team could access it via the Internet from their home institutions. The probe then uplinked data to the fixed server daily via the BGAN satellite connection.

To visualize PLL time series data, new sections were added to the xGDS web interface (Figure 7). The *Plots* section has a menu of 23 scalar time series variables available for plotting. Some variables are available for multiple platforms (both shore station and LakeLander). The user can select which variables to display together, scroll forward or backward in time, and zoom the time resolution in or out to visualize processes at time scales ranging from minutes to months. Scrolling or zooming the time axis of any plot adjusts all the plots in the same way. The *Profiles* section displays color-coded water quality data across time and depth. The user can select the time range to display. To make data gaps visible, sample points are plotted in black.

For more analytical science tasks, an experimental *Visualization Sandbox* section was added to the web site (Figure 8). The sandbox provides a flexible plotting environment similar to MATLAB or Mathematica, but served through the web, so all members of the science team can use it without requiring any local software installation. Users interact with the sandbox by creating and editing shared *notebooks*. Each interactive notebook documents the process of making a set of visualizations—the user enters a plotting command and the resulting plot is immediately displayed beneath it. The current LakeLander database is exposed to the sandbox in an intuitive way, so users generally don't need to worry about importing data and can always regenerate plots with the latest data.

The visualization sandbox is based on several open source Python libraries, including the IPython Notebook,



☐ Live Thu, 13 Dec 2012 13:44:13 ... Thu, 17 Jan 2013 07:21:51 Laguna Negra Local Solar UTC-4:41

Air temperature (C) 08:14:28 **14.560**

Profiler logger minimum battery voltage (V, 1 hour period) 09:19:28 **12.260**

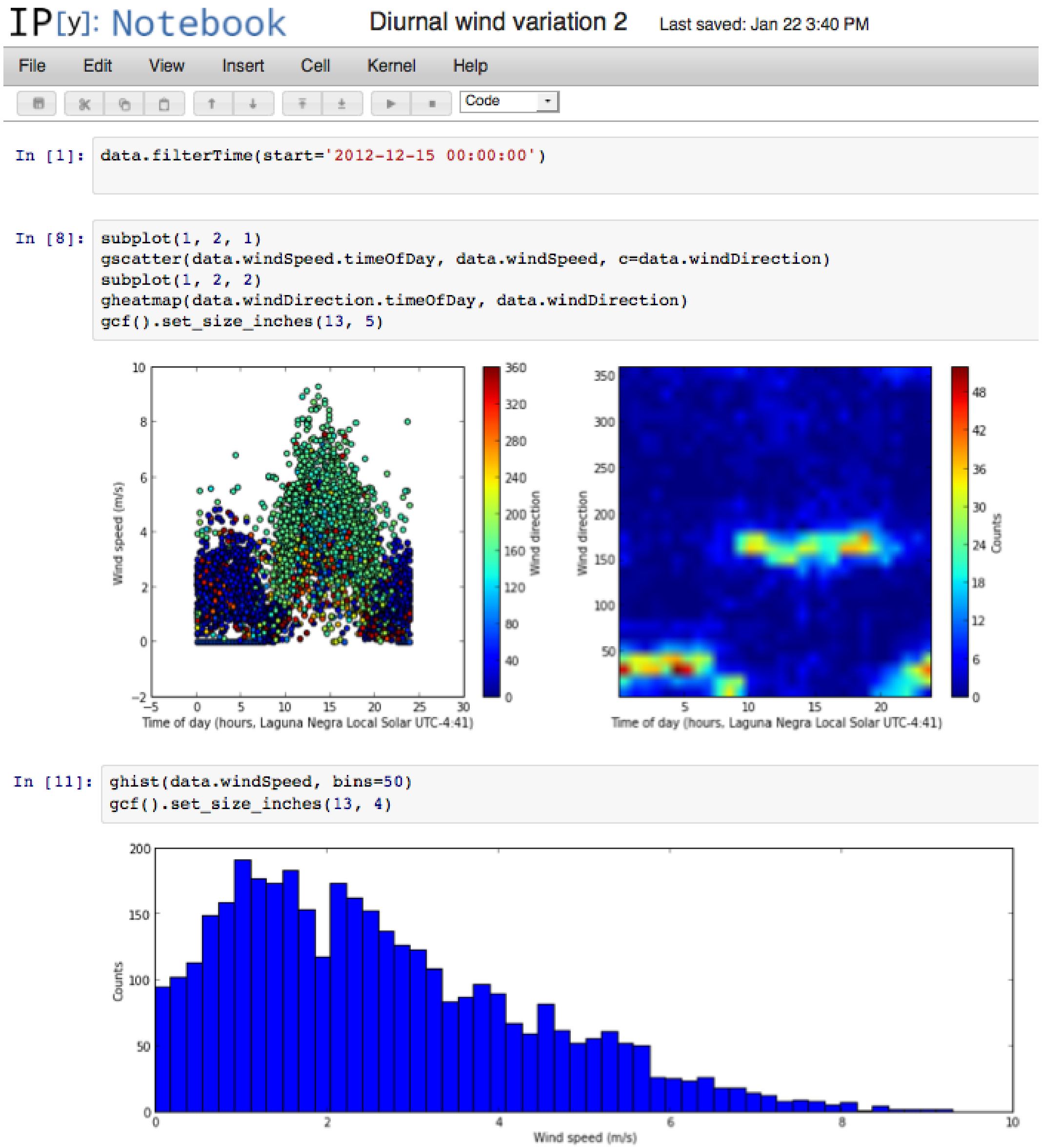
Shore station soil temperature (C) 12:20:28 **27.640**

Time range: 2013-01-11 15:46 .. 2013-01-14 15:46 Laguna Negra Local Solar UTC-4:41 ☑ Show sample points

Figure 7: xGDS time series web interface: (top) Time series scalar plots, showing multiple variables aligned over a common time interval, with highly responsive interactive scrolling, zooming, and selection of which time series to display. (bottom) Water quality profile plots, where water parameters are plotted as a function of time and depth using a color ramp; the user can tune the time range of the plot.

MatPlotLib, and Scientific Python (Pérez and Granger, 2007; Hunter, 2007; Jones et al., 2013). In addition to plotting, these tools provide a wide range of numerical analysis routines. Figure 9 shows an example. The goal was to evaluate a simple ODE model of wind-driven lake level changes—the model attempts to predict variations in the lake level based on observed wind direction and intensity. Working within the notebook, in 35 lines of code, the model is defined, the predicted depth values are generated, and the results are plotted in several different ways. The sandbox is making it possible for scientists to try ideas more quickly and analyze their data more collaboratively.

Maps are a key type of information for almost every field project. When working with an interdisciplinary and international science team, often a project collects map layers with a variety of themes that come from



```
In [1]: data.filterTime(start='2012-12-15 00:00:00')
```

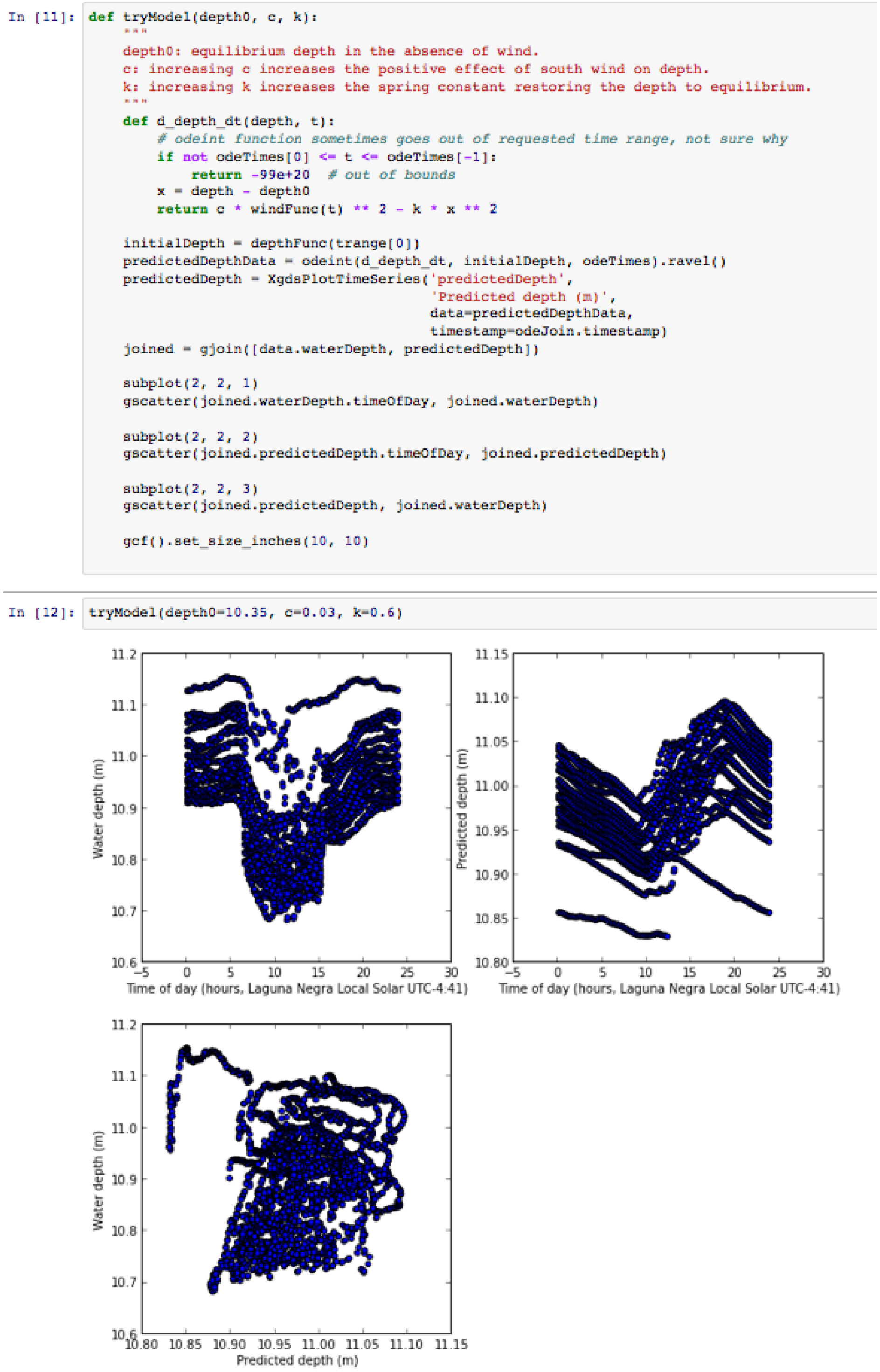
```
In [8]: subplot(1, 2, 1)
        gscatter(data.windSpeed.timeOfDay, data.windSpeed, c=data.windDirection)
        subplot(1, 2, 2)
        gheatmap(data.windDirection.timeOfDay, data.windDirection)
        gcf().set_size_inches(13, 5)
```

```
In [11]: ghist(data.windSpeed, bins=50)
         gcf().set_size_inches(13, 4)
```

Figure 8: xGDS visualization sandbox, showing flexible plotting capabilities in a web-based shell interface.

many different sources. A typical default way to manage this is ad hoc sharing via email, which creates many problems. The team may create many versions of the same map layer (for example, editing a traverse plan), and some team members may inadvertently use an outdated version, causing confusion. Incoming map layers may arrive in different formats, making it difficult to view them together. But probably the most prevalent problem is simply lack of access—the team may have only one or two people who have the appropriate tools and skills to view the map data, greatly reducing its value to the rest of the team.

PLL addressed the map management issue by using the xGDS reusable mapping application (Figure 10), which serves as a shared online repository of the latest version of the map data. Each map layer is converted



```
In [11]: def tryModel(depth0, c, k):
             """
             depth0: equilibrium depth in the absence of wind.
             c: increasing c increases the positive effect of south wind on depth.
             k: increasing k increases the spring constant restoring the depth to equilibrium.
             """
             def d_depth_dt(depth, t):
                 # odeint function sometimes goes out of requested time range, not sure why
                 if not odeTimes[0] <= t <= odeTimes[-1]:
                     return -99e+20  # out of bounds
                 x = depth - depth0
                 return c * windFunc(t) ** 2 - k * x ** 2

             initialDepth = depthFunc(trange[0])
             predictedDepthData = odeint(d_depth_dt, initialDepth, odeTimes).ravel()
             predictedDepth = XgdsPlotTimeSeries('predictedDepth',
                                                 'Predicted depth (m)',
                                                 data=predictedDepthData,
                                                 timestamp=odeJoin.timestamp)
             joined = gjoin([data.waterDepth, predictedDepth])

             subplot(2, 2, 1)
             gscatter(joined.waterDepth.timeOfDay, joined.waterDepth)

             subplot(2, 2, 2)
             gscatter(joined.predictedDepth.timeOfDay, joined.predictedDepth)

             subplot(2, 2, 3)
             gscatter(joined.predictedDepth, joined.waterDepth)

             gcf().set_size_inches(10, 10)
```

```
In [12]: tryModel(depth0=10.35, c=0.03, k=0.6)
```

Figure 9: ODE lake level model in the visualization sandbox. The model is defined, evaluated against probe data, and plotted in a few dozen lines of code.

to a common format (KML) when it is imported, so users can view all of the layers together with the Google Earth map viewer. Users launched the map view by clicking on a web link that loaded the map from the server and launched Google Earth on their computer. In case of changes, the viewer automatically refreshed the map display every hour to show the latest layers. Layers included sample locations, bathymetry, vegetation and geology maps scanned from previous studies of the watershed, topography, landmarks, and multi-spectral remote sensing.

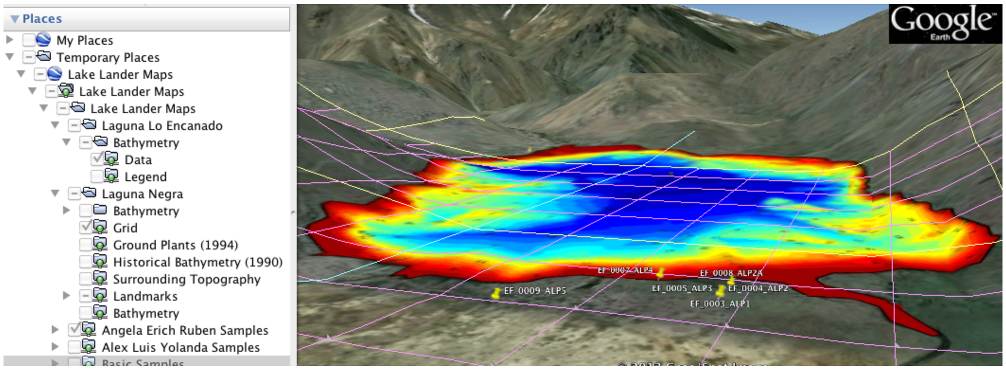


Figure 10: xGDS map display showing sample locations, grid marks, and bathymetry. Bathymetry: Chris Haberle. Map data: MapCity, DigitalGlobe, Inav/Geosistemas IRL, Google.

In future work, the xGDS system will be made available for offline use in the field through a mobile device such as a phone or tablet, termed a "digital field assistant". The primary benefits of this approach are improved management of data collection and improved field safety through shared trip planning and live position tracking. In 2011 and 2012 a preliminary assessment was made of the benefits of live tracking using Garmin Rino GPS-enabled handheld radio transceivers. The units functioned well, were used for most field operations, and got positive feedback from the safety officers.

7 Lessons Learned and Results

The LakeLander robot was first deployed in December 2011 and has operate 13 months to date. It survived the austral midwinter solstice, but required several site visits to correct critical failures of the onboard computer and a subsequent failure of the communications system.

The following timeline summarizes the times when LakeLander was operational on the lake and suffered service outages. LakeLander I refers to the first deployed version of the profiler (without our avionics upgrades to support communication and adaptive science). LakeLander II is the full system after upgrades.

- 2011/11/28: **Field test 1: Deploy LakeLander I**
- 2011/12/15: Lost contact due to blown fuse

- 2011/01/07: Back in service: Blown fuse diagnosed and replaced
- 2012/04/01: LakeLander I shipped back to USA for upgrade
- 2012/11/26: **Field test 2: Deploy LakeLander II**
- 2013/01: Reduced power available: Wind turbine breaks, ops continue
- 2013/04/01: Lost contact due to file system corruption on FitPC
- 2013/04/07: Repair trip: Fix wind turbine, diagnose and retrieve FitPC for repair in USA
- 2013/04/30: Back in service: FitPC replaced with repaired file system
- 2013/06: Reduced profiling capability: Snags in profiler cable limit accessible depth range
- 2013/12/01: **Field test 3: Collect data with LakeLander in motion**

The following section presents lessons learned and results from the field. As the PLL project enters its third year, the focus moves from data collection to data analysis; upcoming publications will help to clarify the data quality and scientific value of these measurements.

7.1 Lessons Learned

Our experience with the Lake Lander yielded several lessons applicable to future projects.

Structuring our platform by layering enhancements on a working integrated COTS system (the YSI profiler) proved to be wise. Starting from the foundation of a weatherized field tested platform greatly simplified the integration task. It enabled testing of the field deployment process and collection of a month of initial data following the first PLL field season in 2011, which would not otherwise have been possible. Nevertheless, the profiler failed in various ways (detailed below). Computer control of the profiler is not directly supported by the vendor, and our unusual use case exposed deficiencies in the vendor hardware and data logger control software that had to be understood and corrected with minimal vendor support.

Extensive measurement, logging, and display of as many system variables as possible has proven key to effective maintenance of a long-lived remote system. Examples of this pattern include: (1) Early lack of instrumentation in the power system meant that educated guesswork was needed to choose which subsystems to turn off in low power modes. (2) Because error messages were not propagated from operating system logs to the engineering team, the probe suffered a system failure that required on-site maintenance, detailed below. (3) Some communications outages were impossible to diagnose in part because the telemetry downlinked before the outage was not sufficient.

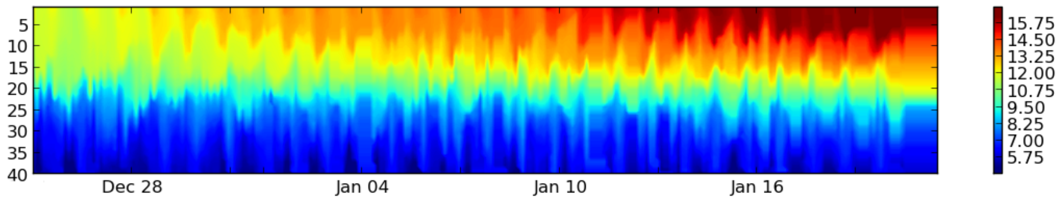
Multiple redundant communication entry points proved vital for robust maintainability of remote systems. Examples include redundant BGAN and Iridium satellite systems, and multiple ways of connecting to the profiler data loggers (via a console interface and a GUI application). This allowed diagnosing faults that occurred when one channel was either not available, or insufficiently capable.

Reducing barriers to accessing and analyzing data broadens interdisciplinary participation by the team. The visualization sandbox component of the xGDS data sharing web site has been one successful example. In one case, the sandbox was used to check for the presence of seiches (standing waves at a resonant frequency of the lake). The analysis quickly determined that (1) there were no detectable seiches in the lake depth time series to date, and (2) modeling predicted that if there were seiches, they would be detectable with a higher sampling rate. In a subsequent maintenance visit, the depth sensor program was modified to sample at a higher rate. The sandbox was then successfully used to find a seiche and calculate its frequency and amplitude. Because the analysis was done in the sandbox, both the raw data and all the processing steps were captured and available to the entire science team. The sandbox has also enabled summer interns to work together and do sophisticated numerical analysis and visualization of incoming probe data with minimal initial training.

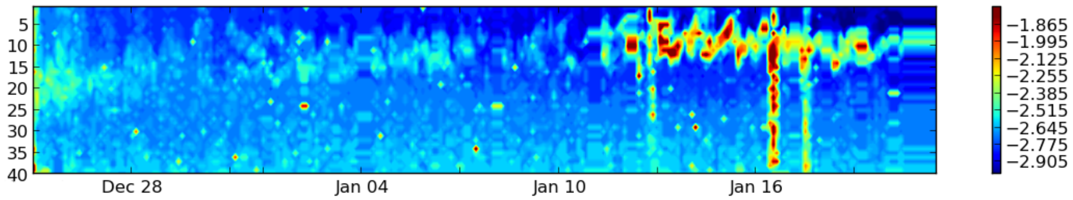
7.2 Data Collected

Figure 11 shows water parameter profiles at various depths plotted against time for a month during the austral summer of 2012/2013. This data was acquired by LakeLander and relayed back to the United States. The science significance of this data set is beyond the scope of this paper, but these plots show that (1) some of the sensors function well and show clear temporal patterns that are scientifically relevant (such as the oscillation of the thermocline depth in the temperature plot), (2) other measurements must be interpreted carefully because the sensors are near their detection limits (such as the turbidity sensor, measuring minuscule opacity changes in an ultra-clear lake), and (3) the plots are drawn from the xGDS visualization of the profile history, which makes it easy for any science team member to see correlations between data sets (users can adjust the time range of the plots).

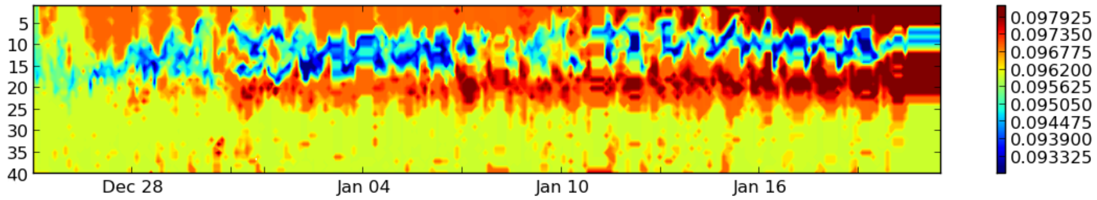
Figure 12 plots wind speed and direction as measured on the profiler. Note the daily cycles of low northerly winds in the morning, followed by higher southerly winds in the afternoon. The wind direction varied significantly at different parts of the lake, but the morning/afternoon shift in direction seemed consistent. These frequent strong wind patterns complicated small-boat navigation on the lake.



(a) Temperature (C)



(b) Turbidity (NTU)



(c) Total dissolved solids (g/l)

Figure 11: LakeLander profiler data. Water quality parameters (temperature, turbidity, etc) from the surface to deep within the water column are plotted against time (Austral summer 2012/2013).

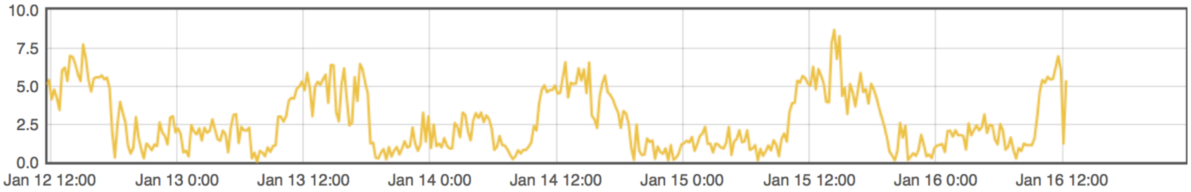
The precipitation sensor on the shore station failed in midwinter, possibly because it was not rated for below freezing conditions (a poor design choice on the part of the vendor).

7.3 Subsystem Outcomes

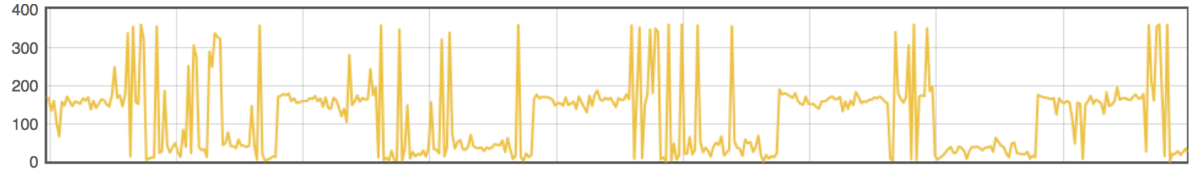
The remainder of this section is a summary of the performance of various subsystems and situations encountered in the field. This includes unanticipated changes the LakeLander system required as well as things that, for better or worse, were surprising.

7.3.1 Scientific Sensors

The commercially supplied profiler and meteorology sensors demonstrated unexpected sensitivities to variations in the power supply voltage. Specifically, the solar powered shore station reports abrupt 20cm changes in lake level that strongly correlate with the sun being shadowed (or not) by the surrounding mountains (Figure 13). Similarly, the profiler reports daily variations *at all depths* in oxidation/reduction potential (ORP) that correlate strongly with daylight hours (when the profiler solar panels are illuminated), superimposed



(a) Wind speed (m/s)



(b) Wind direction (degrees from magnetic North)

Figure 12: LakeLander meteorology data.

over a secular trend.

The lesson drawn is that regulated power supplies should be used for all scientific sensors, even supposedly turnkey commercial units that come with their own power supplies.

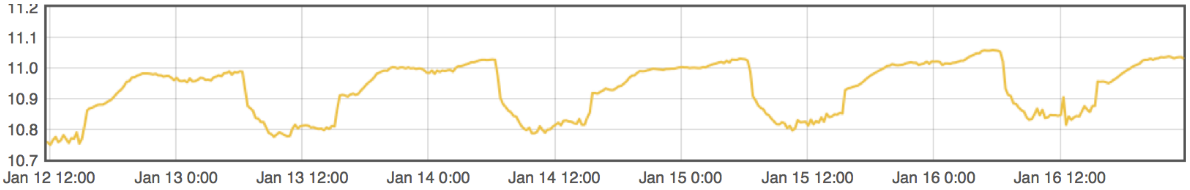
7.3.2 Profiler

The profiler occasionally ceases operations. Observed causes include (1) power supply voltage outside limits, (2) communication glitches between the profiler data logger controllers and the sensor sonde and (3) possible fouling of the cable.

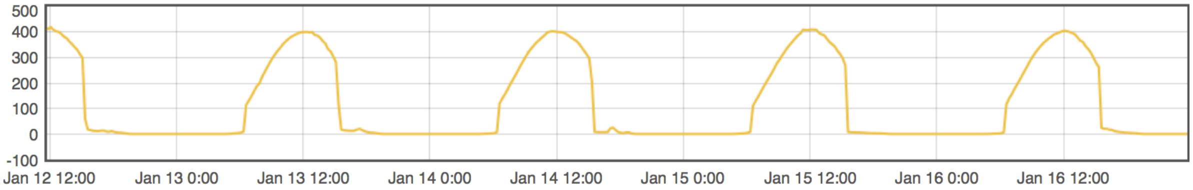
Power supply voltages outside limits are frequent problems that cause the profiler to temporarily stop operating (by design). This limited the probe's ability to obtain profiles throughout the night, when cold temperatures and lack of solar (and wind) power pulled supply voltages below limits.

The communication glitches between the profiler controller and sonde may have been related to battery voltage downward spikes associated with turning the winch motor on and off. They usually resulted in the profiler stopping (waiting for us to restart it) but occasionally cause the profiler to erroneously conclude that the winch is stuck and execute a get unstuck maneuver.

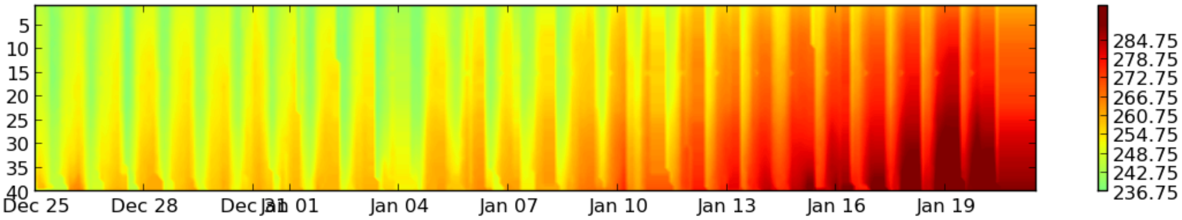
In June 2013 (midwinter) the profiler cable developed an unknown problem that caused the winch to get stuck with the sonde at 13m depth. The profiler's subsequent automated "get unstuck" recovery maneuvers ironically resulted in the cable getting wound in the opposite direction on one occasion, and spooling onto the deck on another. Fortunately, neither event resulted in permanent fouling of the cable, which would have



(a) Water level (m)



(b) Solar insolation (W/m^2) - shore station



(c) Oxidation/Reduction Potential (mV)

Figure 13: Measured lake level (a) and solar insolation (b) at the shore station, showing strong correlation between solar power input and measured lake level, with amplitudes of 20cm. Also shown is measured oxidation/reduction potential (ORP) showing diurnally locked variations at all depths.

required on-site maintenance after roads cleared in the spring. Diagnosing and fixing these events remotely, via satellite, turned out to require IP connectivity between the winch controller and a ground-side Windows PC running vendor-supplied driver software. This validates the approach of connecting most probe avionics via an Ethernet bus.

Since June 2013 the profiler was re-programmed to only profile below 13m to reduce the chance of cable snags. This situation is likely to remain until either (1) winter snows clear sufficiently for someone to visit the sonde and determine what is causing the cable to occasionally snag at 13m, or (2) the automated recovery maneuver can be disabled in the vendor code, reducing the chance that the probe's reaction to an occasional snag will cause permanent fouling.

7.3.3 Power System

The combination of solar panels, wind turbine and power demand management has proven sufficient for LakeLander to acquire up to 12 profiles per day in summer and winter. Figure 14 plots LakeLander's battery voltage over 8 months in 2012. Battery capacity appears to be the main factor preventing the maximum rate of 24 profiles per day.

During summer the solar panels produce sufficient power by themselves to power the system. The wind turbine fell off its mount in late December (Figure 15). Battery voltage levels do not drop until storm related cloud cover occurred 3 weeks later, eventually resulting in a low voltage system shutdown.

During winter a combination of wind power and aggressive demand management (reduced profiling rates and turning the FitPC off for 21 hours per day) maintained battery voltage above the shutdown cut-off despite drastically reduced (1/3) solar insolation levels.

In theory, the original 95 A-hr battery is sufficient to operate LakeLander at the maximum profiling rate for 2.5 days. As the year progressed it became impossible to profile throughout the night, even though battery levels at the end of the day are consistent with a fully charged battery, suggesting that the battery system is significantly degraded. An attempt to increase battery capacity by installing a second lead acid battery in parallel (”borrowed” from a boat) helped marginally, but not to the extent expected.

We suspect that battery degradation is due to a combination of age (discharge cycles), low temperatures and occasionally being overcharged by the wind turbine, but lack the data to confirm this.

The wind turbine was observed to occasionally raise battery voltages above 16V, which caused problems with the profiler. Adding the additional battery partially ameliorated this problem. The wind turbine limits its maximum output voltage by turning off in high winds. Attempts to adjust this voltage resulted in the wind turbine frequently turning off in moderate winds, reducing the daily amount of energy generated.

7.3.4 Computing System

The FitPC embedded system and other electronic components survived the temperature and power fluctuations during the field season. However, one of the most serious system outages, requiring on-site maintenance, was caused by file system corruption related to unclean shutdowns of the FitPC operating system.

Specific lessons can be drawn from this failure: (1) Use of RedHat Enterprise Linux, a distribution designed

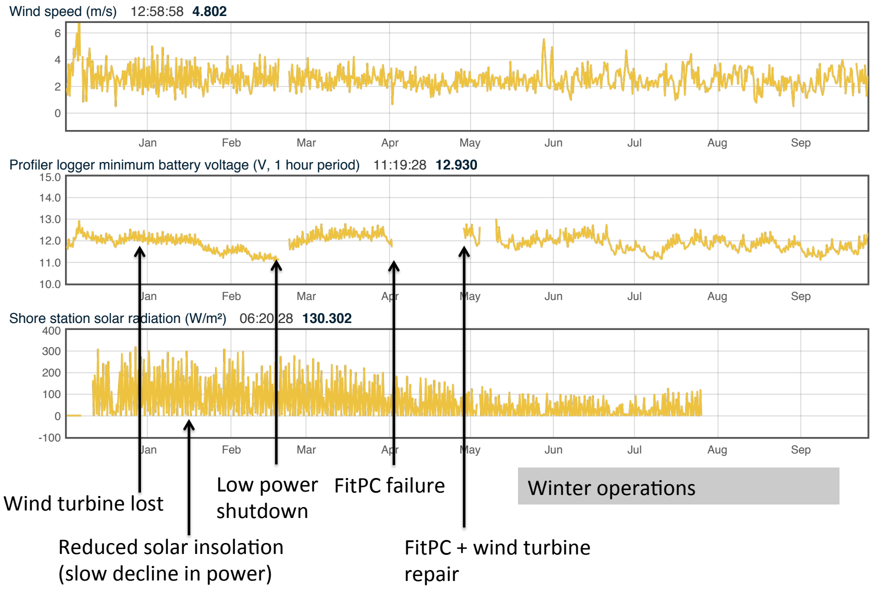


Figure 14: LakeLander power system performance (2012) showing effects of (1) high winds, (2) wind turbine loss, (3) overcast conditions and (4) winter low sun angles on the LakeLander battery voltage.

Figure 15: The LakeLander wind turbine was lost in late December 2012 due to vibrations causing initially tight screw fittings in the support structure to unwind.

for server and desktop computers (a choice motivated in part by administrative constraints in our workplace), introduced unnecessary complexity that tended to hide problems in the boot and shutdown sequences. A stripped-down distribution intended for embedded computers would have been more appropriate. (2) The choice of storage hardware, partitioning, and file system must be made carefully for best reliability. Critical file systems involved in the boot process should be mounted in a read-only configuration to prevent corruption even in a sudden power-loss situation. Partitions that must be writable (e.g. for logging data) should use a robust journaling file system and mount settings that favor reliability over maximum write performance.

But a larger point is that the unclean shutdown behavior started weeks before the system failure, and could easily have been corrected before it eventually caused file system corruption, if code had been in place to monitor the appropriate system logs and propagate warnings to the attention of the engineering team. A recurring theme in system design is that in order to avoid being hit one must be aware of the near misses.

7.3.5 Satellite Communications

The BGAN satellite communications system has proven reliable since December 2012. Over the past 75 days since formal availability logging was started, the probe has been able to ping the USA-based server over the BGAN link on 74 days (99%), and had a sufficiently stable connection to initiate the daily data uplink on 69 days (92%).

There were concerns about pointing a fixed-directional antenna at a geostationary satellite from a float platform. In part to address this concern, the pontoon is moored from both ends by extra-heavy duty anchors (50 kg apiece plus 15 kg of heavy bottom chain) that keep it in place and (so far) keep yaw variations within 15 degrees, sufficient for keeping the antenna pointed correctly. Wave induced rocking motion does not appear to be a factor in this lake.

The Q-Pro Iridium modem was installed as backup for the BGAN. When the Q-Pro is powered up, messages in either direction usually get delivered in minutes. However, messages sent to the unit when it was unpowered or not registered would sometimes take days (or be dropped altogether). The solution to this is to keep the unit on continuously (when the FitPC is on), and check for messages every 5 minutes. In addition, duplicate and old command messages sent to LakeLander via the Q-Pro are ignored so that a message can be sent multiple time to assure receipt.

The legacy 9522B Iridium modem connection to the data loggers has proven to be unreliable. Iridium satellites are usually low on the horizon, and therefore frequently occluded by the mountains surrounding Laguna Negra. Continuous Iridium data links could rarely be maintained for longer than 5 minutes, with downtimes of approximately 10 minutes in between. Connection times of at least 10-20 minutes are required for the daily profiler data alone. At $1.50 per minute, this equates to $15-$30 per day, not competitive with BGAN. Connections are further complicated by the scarcity of analog phones and modem traffic blocks put in place by our home institution (NASA Ames Research Center), and multiple tries required to establish a connection.

The Iridium modem worked well enough to download data every 2-3 days in December to January 2012, after which connections initiated from an analog landline could no longer be established. We were unable to determine the cause despite considerable effort. Connections originating from a mobile Iridium handset remained possible, albeit not practical for operations (requiring a terminal with a clear view of the sky).

7.3.6 Local Communications

The 900MHz Microhard radios proved reliable when LakeLander was moored directly in front of the base camp. Communications rates of 230kbps allowed easy access to the machine (via `ssh`) and enabled local testing and development, since the device was only physically accessible by boat during the day if visibility was good and average wind speeds were below 5 m/s. Once installed at its final mooring place, 3.3 km from base camp, signal strength dropped dramatically and the link became unreliable (occasional complete outages and up to 80% packet loss). The link quality was improved slightly by increasing signal power and adding an additional mooring line to keep a nearby bluff out of line of sight. More robust communications were possible from an area to the west of base camp with better line of sight to the system.

The power regulator for the Microhard unit left on site in 2011 failed after about 2 months operating at maximum power. The replacement radio, operating at 1/5 of maximum power, has so far operated reliably.

7.3.7 Pan-Tilt-Zoom Imager

The pan-tilt-zoom camera used for acquiring high resolution image panoramas was initially mounted in a standard camera facing down configuration for a view of the robot and surrounding water. This proved inadequate for imaging the nearby waterfalls and surrounding terrain, so it was flipped over (with a small forward angle to maintain a view of the falls). In this configuration the camera, which is inside a clear plastic dome, is no longer shaded from the noon sun, and appears to be overheating occasionally when used at midday (air temperatures average 20C). During in-person testing, we have observed that when the camera overheats, it outputs images degraded with streaks, and that behavior can be corrected by opening the dome to allow air flow. When the camera is unattended, we infer that similar streaks are likewise due to overheating.

7.3.8 Thermal System

CPU temperatures logged when the FitPC is active have never fallen below freezing, even with sub-zero air temperatures (Figure 16). We infer the SSD temperature is likewise above freezing and within its operating temperature range.

This success may be due to the avionics box insulation successfully retaining enough daytime heat overnight. Another possibility is that in sub-freezing weather the avionics are very briefly exposed to sub-freezing temperatures at boot, with some attendant risk, but are sufficiently well-enclosed that they are already warm by the time the FitPC boot sequence gets far enough along to begin internal temperature logging. In either case, we have not observed any avionics problems we attribute to thermal conditions.

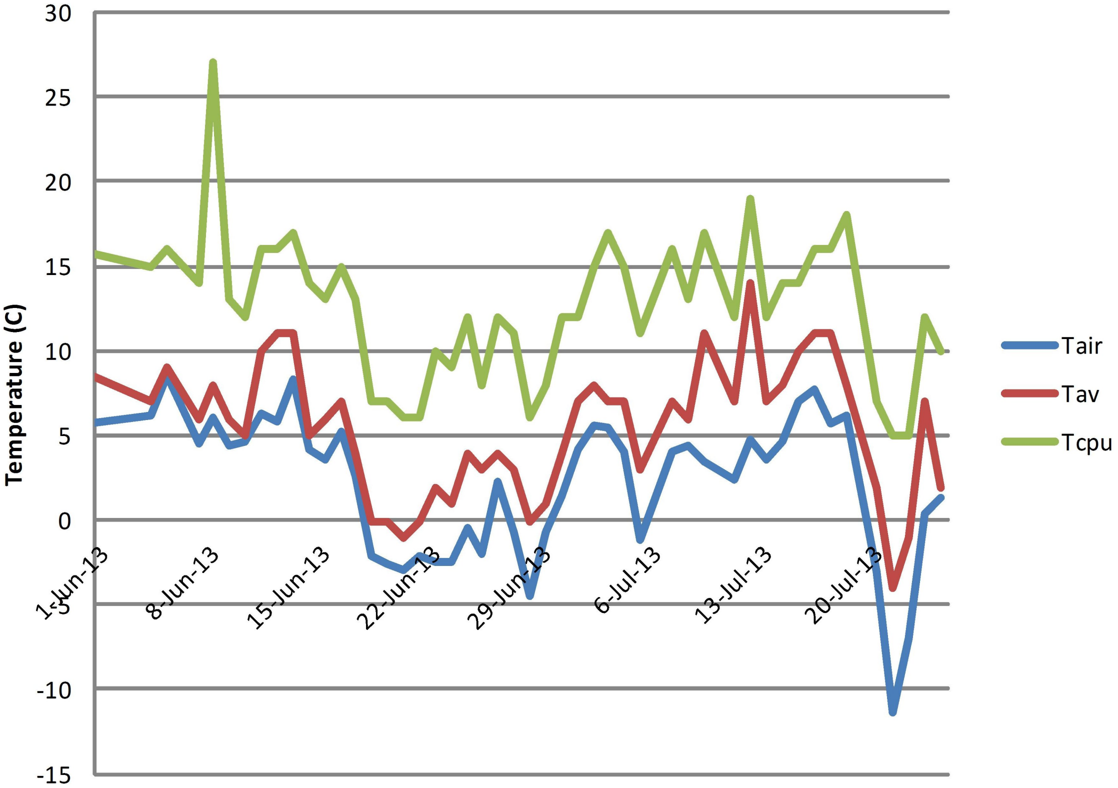


Figure 16: Wintertime thermal system performance. FitPC CPU temperature (T_{CPU}) stayed consistently above the avionics box temperature (T_{av}) and external air temperature (T_{air}), and within the operating temperature range of the SSD. All temperatures are daily values logged at FitPC boot.

8 Conclusions and Future Work

This paper discussed design trade-offs and field outcomes for Lake Lander's systems, with the goal of informing the design of similar future projects. To date, the Planetary Lake Lander Project has addressed Dunbabin and Marquez's (2012) themes of reliability, safety, endurance, and human-robot interfaces.

Statistically unusual event detection and process tracking are necessary for LakeLander to capture data at sufficient rates during periods of high variability. LakeLander's data collection rate is limited by available energy, particularly during winter months when no more than 4 profiles per day were possible. Energy consumption is dominated by the winch motor. Short of overhauling the power supply (more battery capacity, better charge controllers and an additional DC-to-DC power regulator to maintain voltage while discharging the batteries further), we conclude that the best option to assure high rate (hourly or better) profiling is to concentrate profiling activity around periods and depths of interesting activity.

The proposed Titan Mare Explorer (TiME) mission has even tighter power and communications constraints (400 MB over 6 months, costing 1 joule per bit transmitted; Stofan et al., 2010). Onboard selection of the most interesting data to be collected and uplinked is likely to prove significant for missions such as TiME.

Future work with LakeLander is to develop the onboard adaptive science system for unusual event detection (e.g. storms, discharges into lake, algal blooms) and process tracking to focus measurements, prioritize data downlink and alert users to unusual activity.

8.1 Acknowledgments

This work is funded by NASA's Astrobiology Science and Technology for Exploring Planets (ASTEP) program. Additional expertise was drawn from the Intelligent Robotics Group at NASA's Ames Research Center, specifically Mark Micire, Matt Deans, David Lees, Tamar Cohen, and Vinh To.

References

Cabrol, N. A. and Grin, E. A. (2010). *Lakes on Mars*, volume Declining Lake Habitat in Rapid Climate Change, page 408. Elsevier.

Cabrol, N. A., Grin, E. A., Chong, G., Minkley, E., Hock, A. N., Yu, Y., Bebout, L., Fleming, E., Häder, D. P., Demergasso, C., Gibson, J., Escudero, L., Dorador, C., Lim, D., Woosley, C., Morris, R. L., Tambley, C., Gaete, V., Galvez, M. E., Smith, E., Uskin-Peate, I., Salazar, C., Dawidowicz, G., and Majerowicz, J. (2009). The High-Lakes Project. *Journal of Geophysical Research*, 114(28):1–20.

Cabrol, N. A., Grin, E. A., Haberle, C., Moersch, J. E., Jacobsen, R. E., Sommaruga, R., Fleming, E. D., Detweiler, A. M., Echeverria, A., Blanco, Y., Rivas, L. A., Pedersen, L., Smith, T., Wettergreen, D. S., Demergasso, C., Parro, V., Fong, T., Chong, G., and Bebout, L. (2012). Planetary lake lander: Using technology relevant to Titan's exploration to investigate the impact of deglaciation on past and present planetary lakes. In *Proc. Lunar and Planetary Sci. Conf. (LPSC).*

Carpenter, S. R. and Brock, W. A. (2006). Rising variance: A leading indicator of ecological transition. *Ecol. Lett*, 9(3):311–318.

Deans, M. C., Smith, T., Lees, D. S., Scharff, E. B., and Cohen, T. E. (2013). Real-time science decision support tools: Development and field testing. In *Proc. Lunar and Planetary Sci. Conf. (LPSC).*

Dunbabin, M. and Marques, L. (2012). Robotics for environmental monitoring. *IEEE Robotics and Automation Magazine*, 19(1).

ESA/NASA (2009). Titan Saturn system mission. In *NASA/ESA joint summary TSSM Report*, page 32.

Flückiger, L., To, V., and Utz, H. (2008). Service-oriented robotic architecture supporting a lunar analog test. In *In Int. Symp. on AI, Robotics, and Automation in Space (iSAIRAS).*

Hitz, G., Pomerleau, F., Garneau, M.-E., Pradalier, C., Poschy, T., Pernthaler, J., and Siegwert, R. (2012). Autonomous inland water monitoring - design and application of a surface vessel. *IEEE Journal of Robotics and Automation*, 19(1).

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing In Science & Engineering*, 9(3):90–95.

Jones, E., Oliphant, T., Peterson, P., et al. (2001–2013). SciPy: Open source scientific tools for Python.

Lee, S. Y., Lees, D., Cohen, T., Allan, M., Deans, M., Morse, T., Park, E., and Smith, T. (2012). Reusable science tools for analog exploration missions: xGDS web tools, VERVE, and Gigapan Voyage. *Acta Astronautica.*

Pérez, F. and Granger, B. E. (2007). IPython: a System for Interactive Scientific Computing. *Comput. Sci. Eng.*, 9(3):21–29.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foot, T., Leibs, J., Wheeler, R., and Ng, A. (2009). ROS: An open-source Robot Operating System. In *ICRA Workshop on Open Source Software.*

Stofan, E. R., Lorenz, R. D., Lunine, J. I., Aharonson, O., Bierhaus, E., Clark, B., Griffith, C., Harri, A.-M., Karkoschka, E., Kirk, R., Kantsiper, B., Mahaffy, P., Newman, C., Ravine, M., Trainer, M., Waite, H., and Zarnecki, J. (2009). Titan Mare Explorer (TiME): First in situ exploration of an extraterrestrial sea. In *Presentation to the NASA Decadal Survey.*

YSI (2012). YSI buoy based vertical profiler. http://www.ysisystems.com/systemsdetail.php?Buoy-Based-Vertical-Profiler-10.