Loon Library

Lessons from Building Loon's Stratospheric Communications Service

Loon Library: Lessons from Building Loon's Stratospheric Communications Service by Aswin Alexander, Marc Alvidrez, Wajahat Beg, Zoe Benezet-Parsons, Léonard Bouygues, Scott Coriell, Jameson Dempsey, Umit Dogruer, Ben Freedman, Jon Grazer, Derek Herbert, Bryan Ho, Rick Lange, Marc Lelièvre, Don Nguyen, Jonathan Nutzmann, Ken Riordan, Kevin Roach, John Rousseau, Robert Schlaefli, Sam Truslow, and Paul Heninwolf on behalf of the Project Loon team.

Additional information: https://x.company/projects/loon

Published August 2021

Loon Library
Lessons from Building
Loon's Stratospheric Communications Service

CONTENTS

Chapter 1 Loon Overview and Context

Chapter 2 System Engineering

Chapter 3 Flight Vehicle

Chapter 4 Communications Systems and Service Design

Chapter 5 Loon Software

Chapter 6 Engineering Production Software and Integrated Systems

Chapter 7 Interface with the Real World

Chapter 8 Loon's Trajectory and Future Impact

Chapter 1

Loon Overview and Context

Most projects born at X, the Moonshot Factory, begin by trying to prove that a technology will not work. Healthy skepticism from the team charged with building an audacious technology is a fundamental aspect of building one that is scalable and sustainable. As projects advance and breakthroughs are made, skepticism naturally gives way to optimism that the technology just might be capable of the potential imagined for it.

An Introduction to Loon

Over its nearly ten year journey, this was the trajectory that Project Loon took. What initially seemed fanciful — flying giant internet balloons to connect users in hard-to-cover areas around the world — eventually became possible. By the end of its journey, Project Loon successfully connected hundreds of thousands of users worldwide with networks of floating cell phone towers operating in the stratosphere.

But technical feasibility does not always directly correlate to a sustainable business model. It is almost impossible to determine if the correlation exists at the beginning of a project. For assumptions to be tested, technology must be built, breakthroughs are necessary, and business risks must be taken. Loon did all of these things, but unfortunately, the road to commercial viability proved much longer and riskier than hoped.[1]

1 https://blog.x.company/loons-final-flight-e9d699123a96

This is the Loon Library. Contained in these pages are the technical innovations and key learnings that the Loon team has gathered throughout the decade-long journey. Loon shares the accumulated knowledge and innovations to assist others who might build upon Loon's work in aviation, communications, and artificial intelligence. The authors hope that the Loon Library contents will enable others to advance Loon's mission to connect the unconnected and eliminate the digital divide.

Loon's Mission

The Internet feels ubiquitous, but it's not: nearly half of the world's population lacks affordable access to basic connectivity. Loon sought to bridge the divide by pioneering high-altitude communications technology that makes it possible to connect more people, places, and things worldwide. From the early days of testing weather balloons, designing and building custom launch equipment, providing connectivity to people after natural disasters, and ultimately bringing the Internet to hundreds of thousands of people worldwide, Loon was committed to tackling the challenge of connecting the unconnected.

Throughout the nearly decade-long pursuit, Loon made tremendous progress towards this mission through innovation, commitment, and partnership. However, the journey proved to be much longer and riskier than hoped, and Loon made the difficult decision to wind down operations.

How Loon Works

Cell towers in the sky is a simple yet appropriate description of Loon's solution. A network of balloons, each carrying the equivalent of a cellular base station, could connect to each other, and to specialized ground gateway equipment, to provide cellular coverage to users on the ground. All the user needed was a standard LTE mobile phone, just like the one used to connect to a traditional terrestrial network. Figure 1-1 shows the Loon network architecture.

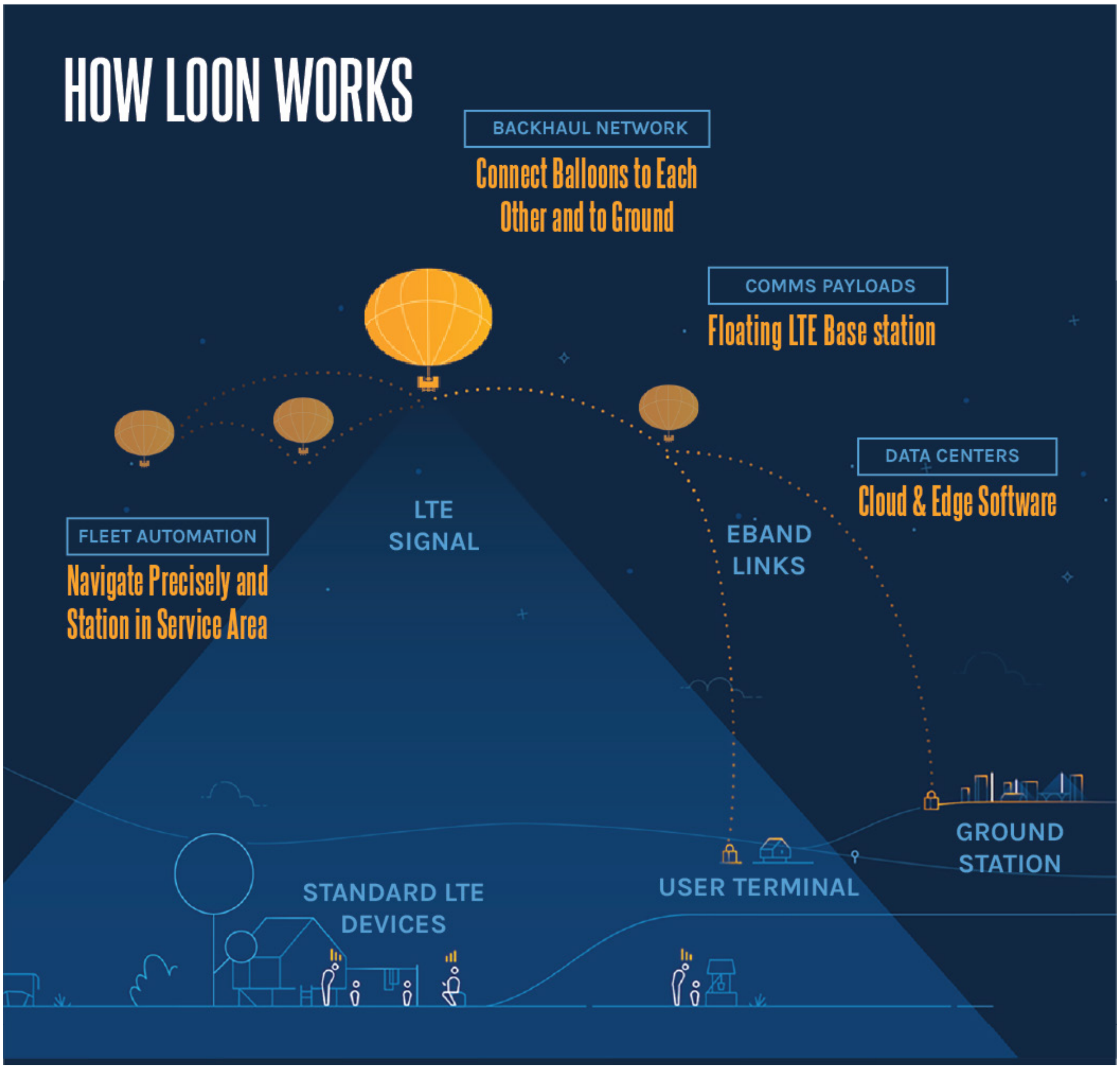


Figure 1-1 *How Loon works.*

Loon's connectivity solutions were designed so that mobile network operators could extend the reach of their networks to attract new customers, while adding resiliency to those networks to better serve existing customers in times of natural disaster or outage. The solutions helped mobile network operators unlock new connections with a flexible and future-proof solution and enabled operators to expand their coverage to unserved and underserved areas with a fleet of stratospheric flight systems equipped with LTE base stations.

Users were able to use standard LTE-enabled handsets to connect to the operator's network via Loon's floating cell phone towers. Figure 1-2 shows the launch event for Loon service in Kenya. See the full video here: First Loon Call in Kenya.[2]

Figure 1-2 The Kenya ICT minister conducts a video call with the President of Kenya over the Loon network.

Loon's adaptive, resilient fleet could accommodate an easily upgradable telecommunications payload, while providing coverage that was easily expandable. Deployed with a Network-as-a-Service model, the fleet was able to coexist with, and complement terrestrial networks, and could be repositioned for maximum impact. Loon's standard LTE technology used the operator's existing spectrum so as to simplify licensing concerns and user device requirements. Loon was able to maximize value by delivering seamless connectivity to subscribers through a unique solution of ground gateways, flight vehicles and software.

2 https://youtu.be/3c4ZxZ-1iuY

Loon Program History

Loon's earliest tests began back in 2011, using a weather balloon and basic, off-the-shelf radio parts to create the first prototype. The next two years were a process of rapid iteration to prove that balloon-powered internet might just work. By 2013, a Loon balloon had completed a lap around the world in 22 days, and Loon's balloons had travelled 500,000 kilometers. These learnings led to major improvements in wind prediction models, balloon trajectory, forecast, and navigation.

Figure 1-3 Prototype Loon balloons ready to manually launch in New Zealand in 2013.

It was also in 2013 when a sheep farmer in Canterbury, New Zealand became the first person to connect via balloon-powered internet through an antenna attached to the roof of his home. That same year, Loon was revealed to the public, which helped to explain some UFO sightings that were reported after test flights around the world.

In 2014, a local school in Agua Fria, in the rural outskirts of Campo Maior, Brazil was connected to the internet for the first time through a balloon launched nearby. This marked Loon's first successful LTE connection. In 2015, The team developed a customized launch system that can fill, lift, and launch our tennis-court sized balloons in under 30 minutes.

Figure 1-4 ***Early Loon fixed wireless service.***
Installing terminals on roof of school in Agua Fria, Brazil.

The year 2017 marked the first major public deployments of Loon service. Working together with Telefonica and the Peruvian Government, Loon delivered basic connectivity to tens of thousands of people in flood-affected areas across the country. This also marked the introduction of LTE technologies provided by Nokia into the Loon fleet. Only weeks after the Peru deployment, Hurricane Maria brought devastation to the Caribbean. Collaborating with AT&T and T-Mobile, the Federal Communications Commission, the Federal Aviation Authority, FEMA, and many others, Loon provided basic connectivity to 200,000 people in Puerto Rico. Loon launched balloons from Nevada, and using advanced navigation software, harnessed stratospheric winds to direct them over Puerto Rico.

Figure 1-5 ***Loon on other aircraft.***
Fitting the Loon payload on a Sunglider aircraft.

By 2019, the achievements of Loon had become well known. Consequently, other companies with similar or complementary missions emerged as partnership opportunities. Softbank's HAPSMobile and Loon formed a long-term strategic relationship to advance the use of high-altitude vehicles, such as balloons and unmanned aircraft systems to bring connectivity to more people, places, and things worldwide. Loon and HAPSMobile went on to develop a communications payload for the Sunglider[3] (an unmanned, fixed wing aircraft), demonstrating the versatility of Loon's technology.

Loon's custom-built temporospatial Software Defined Network (TS-SDN) was selected by Telesat, a leading global satellite operator, to support the development of their next generation low earth orbit (LEO) global communications satellites. Also in 2019, Loon successfully achieved over 1 million hours in the stratosphere[4], a harsh and hostile operating environment. Along the way, Loon traveled 70 million kilometers (enough to make 100 trips to the moon).

3 https://en.wikipedia.org/wiki/HAPSMobile#Hawk30
4 https://medium.com/loon-for-all/1-million-hours-of-stratospheric-flight-f7af7ae728ac

Figure 1-6 ***Loon fleet over Peru, Bolivia and Brazil.***
Loon fleet management system dashboard showing mesh backhaul network serving LTE users in Loreto region.

By 2020, Loon had worked closely with Telefonica[5] to test coverage of Loon's stratospheric network in the rural Loreto region of Peru and remote parts of the Amazon. This relationship unlocked significant technical milestones for Loon's communications and aviation systems, and also provided real-world experience working with a range of government partners on this new frontier of connectivity. In May of the same year, Loon and AT&T[6] announced a partnership designed to respond more quickly and effectively to disasters worldwide. The year 2020 also marked the start of Loon's first commercial network deployment with partner Telkom Kenya[7], pioneering the use of high-altitude balloons to provide LTE coverage across large areas in Kenya. Additionally, Loon and Vodacom[8] signed an agreement in 2020 to bring Loon balloon-powered internet to the South African mobile operator's network in Mozambique.

---

5 https://medium.com/loon-for-all/loon-signs-deal-to-bring-balloon-powered-internet-to-amazon-rainforest-region-in-peru-34696714976c
6 https://medium.com/@awestgarthloon/working-with-at-t-to-offer-a-global-connectivity-solution-in-times-of-disaster-450d8cb9a448
7 https://medium.com/@awestgarthloon/loon-is-live-in-kenya-259d81c75a7a
8 https://medium.com/@awestgarthloon/bringing-loon-to-mozambique-1a0aea489b74

Loon Technical Accomplishments Summary

To put Loon's technical accomplishments into proper perspective, consider the major challenges involved. The stratosphere is a harsh, ever-changing environment. Taking advantage of the inhospitable environment requires overcoming several key challenges, including:

- Keeping the platform working properly in the stratosphere for hundreds of days
- Getting the system to stay over the people who need service
- Coordinating a fleet of multiple balloons that are constantly moving with respect to each other, and with respect to users below

For the communications platform, Loon took the essential components of a cell tower and redesigned them to be light and durable enough to be carried by a balloon 20 km up, on the edge of space. Loon balloons were designed and manufactured to endure the harsh conditions in the stratosphere, where winds can blow over 100 km/hr, temperatures can drop as low as -90°C, and ultraviolet radiation can inflict significant stresses on materials. Despite these challenges, Loon's technical advances repeatedly broke flight duration records, consistently reaching balloon lifetimes over 300 days and the latest generation on track for longer than one year lifetimes.

From a fleet steering perspective, Loon designed, built and evolved an automated fleet operation system that eventually leveraged machine learning and artificial intelligence advancements. Highly skilled flight engineers supervised the system to provide human oversight 24 hours a day, seven days a week, 365 days a year. Loon flight systems were able to reach countries around the world from launch sites in the United States. Rather than being at the mercy of the winds at a single altitude, Loon's systems could ascend and descend to find and select a series of wind currents to navigate to destinations on the opposite side of the world. Predictive models of the winds and autonomous decision-making algorithms helped these flight systems continuously find suitable wind patterns to carry out their missions efficiently. Accurately and safely operating the fleet required understanding how the weather, various flight path restrictions, and flight systems interact over hours, days, weeks, and months. By 2018, Loon was simulating up to 30 billion balloon days every calendar day to anticipate dynamic changes to current operations, to train algorithms and to asist in the design of future vehicles or potential service areas.

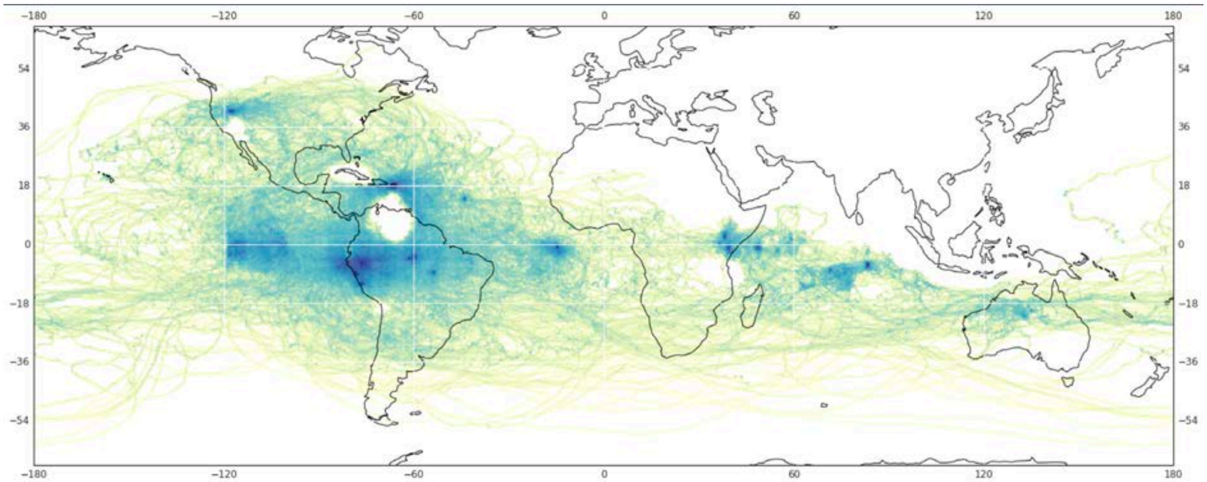


Figure 1-7 ***Loon flights.***
Almost 2 million hours and more than 70 million km worldwide.

A network of multiple Loon balloons is constantly moving. To contend with this situation, Loon developed ground-breaking network orchestration software to manage the dynamic network topology. Loon software worked continuously to choreograph dozens of balloon-to-balloon and balloon-to-ground interconnections to enhance service availability and network performance. This software orchestration system was completely autonomous, efficiently routing connectivity across balloons and ground stations while considering balloon motion, obstructions, and weather events. Each Loon balloon could connect simultaneously to the ground and other nearby flight vehicles. By linking multiple balloons together, Loon created a redundant mesh network that provided resilient coverage over vast geographic areas.

Ongoing Efforts in Industry and Academia

In 2020, Loon united with other leaders[9] across the telecommunications and aerospace industries to create the HAPS Alliance[10], an organization designed to collectively advocate for High Altitude Platform (HAPS) technology and business development. HAPS Alliance is working to create a cooperative HAPS ecosystem, develop standard product specifications and promote the standardization of HAPS network interoperability. After only one year since its inception, the HAPS Alliance counts over 40 organizations among its membership.

Despite Loon's departure, the Alliance remains well positioned to drive HAPS technology towards wide-scale commercial adoption. In addition to industry leaders from both the telecommunications and aviation industries, the Alliance also counts several renowned academic institutions amongst its membership. This cooperation between industry and academia will help drive the continued development of HAPS technologies.

The Stratosphere

The stratosphere is the second layer of Earth's atmosphere, lying above the troposphere. The bottom and top edges of the stratosphere are not arbitrary altitudes but are defined by how the atmosphere's temperature changes as you go higher. From ground level and up through the troposphere, the temperature decreases with increasing altitude but, at some point, the temperature levels out and then starts to increase with increasing altitude. This is the start of the stratosphere, which continues for about 40 km. At that point, the warming trend reverses, with the temperature again getting colder at higher altitudes. The altitude at which the temperature gradient changes direction depends on the season and the latitude. The stratosphere typically starts between 7 km and 18 km altitude and extends to about 50 km. See Figure 1-8.

Compared to lower altitudes, where all but a few aircraft fly, the stratosphere around 20 km has seen very little commercial use. However, over the past few decades, there have been increasing proposals for providing services from these altitudes. Examples include providing communication services and earth observation services that use exceptionally long flight duration vehicles such as airships, high-efficiency fueled fixed-wing aircraft, and solar-powered fixed-wing aircraft.

9 https://medium.com/loon-for-all/telecom-technology-and-aviation-industry-leaders-join-forces-to-create-the-haps-alliance-2af43492fc08

10 https://hapsalliance.org/

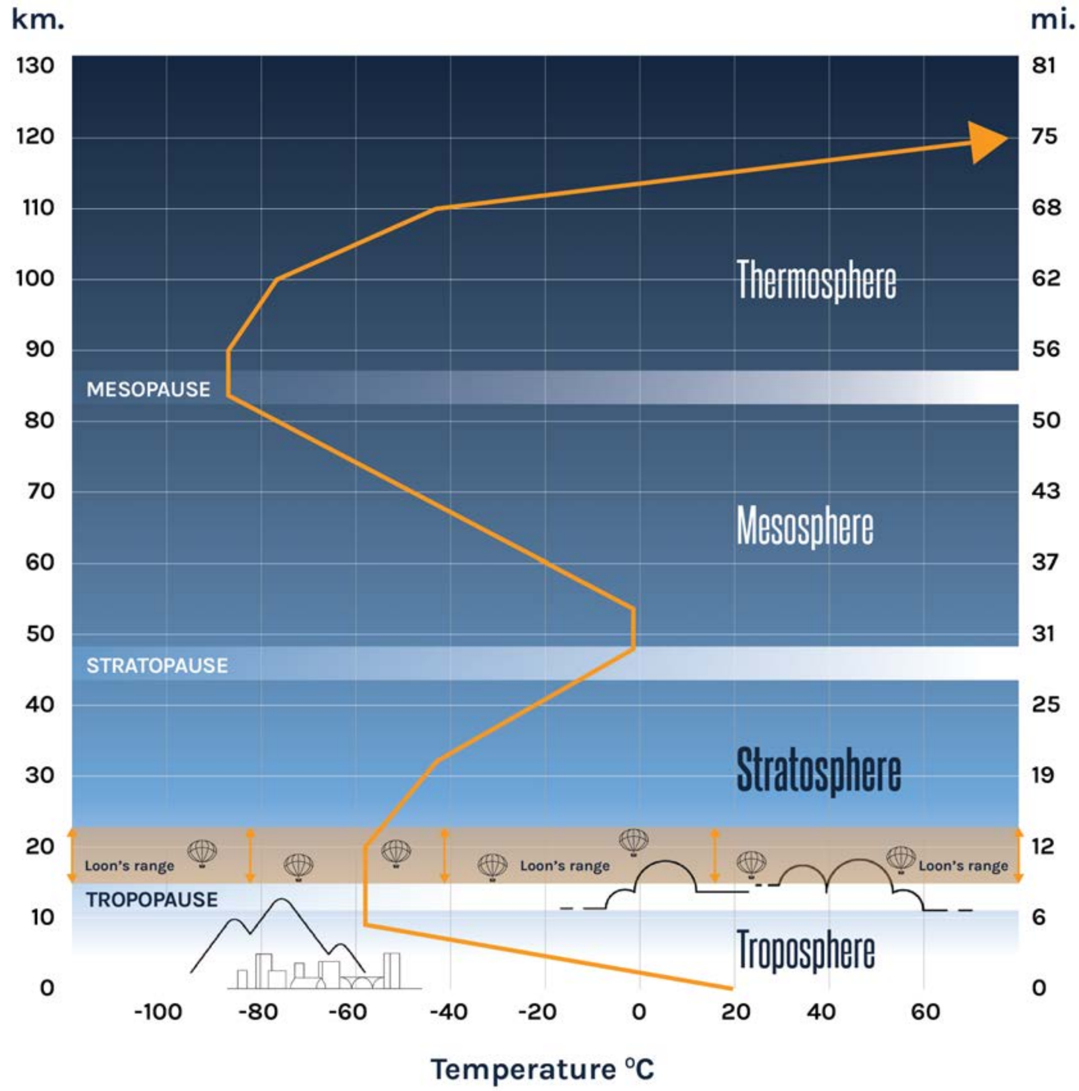


Figure 1-8 ***Stratosphere is defined by temperature increase with increasing altitude.***

Advantages and Challenges in Operating from the Stratosphere

Commonly discussed benefits to operating in the stratosphere are:

- Large operating footprint per vehicle vs. terrestrial or lower-altitude aircraft
- Better radio performance and better image resolution vs. satellites
- Minimal weather: no precipitation, low turbulence, and low electrical activity
- No commercial aircraft sharing the airspace
- Lower cost per service area vs. satellites

Being above typical aircraft altitudes (up to ~51,000 feet or ~15 km) but well below outer space (above 300,000 feet or 100 km), the stratosphere shares advantages and disadvantages of both. It is significantly lower than satellites fly, so telecommunications services can use lower transmission power and smaller antennas. Or, with the same size of equipment and power, significantly higher data rates can be delivered to more people per given area vs. satellites.

Far above the altitude of even the tallest terrestrial towers, a single stratospheric vehicle could potentially cover a service area that would require dozens to hundreds of communications towers. It was this benefit that inspired the original Loon vision.

The stratosphere can use smaller, less expensive earth and environmental monitoring or imaging services than satellites require. Alternatively, instrumentation like those flown on satellites can provide significantly higher resolution. Being approximately twice the altitude of commercial flights, such imaging and environmental monitoring services could cover four times the area of standard aircraft, again, for similar instruments. Loon did not pursue anything in this space.

In addition, being above all the storms and precipitation found in the troposphere, vehicles in the stratosphere do not need to contend with moisture, massive turbulence, or direct lightning strikes. That said, there's more turbulence and more electrical activity there than many think.

Operating in a relatively empty region well above the traditional commercial airspace provides stratospheric vehicles with fewer constraints on their operations. Still, overflight, launching, and landing do require permission from the local governments.

Despite these advantages, operating in the stratosphere has some significant challenges, mainly due to the extremely low air density, which has enormous implications for the flight vehicle design and cost. Plus, the extreme thermal environment, which affects all electronics and mechanisms. The thermal environment shares some of the challenging problems that satellites need to contend with, including extremely low temperatures, significant day/night variations, and little to no convection. Most of the heat dissipation needs to happen through radiation.

The result of the extremely low air density (approximately 1/10th that of sea level and 1/3rd that of commercial airspace) means that fixed-wing aircraft get little lift from their wings and minimal thrust from their propellers. It also means that balloons must be much larger to lift the same amount of mass. All of this corresponds to significantly more expensive vehicles than initially might be considered.

Overview of Long Endurance Stratospheric Vehicles

All the vehicles that have been proposed for continuous or near-continuous service from the stratosphere have been autonomous. They require no pilot in the aircraft nor remote pilot on the ground. This, plus the vehicles' exceptionally long flight times, leads to significant cost reductions. Generally classified as High-Altitude Long Endurance (HALE) vehicles, the flight times range from several days to many months to reduce the operational costs associated with landing, refueling, and re-launching.

The autonomous vehicles proposed for stratospheric operations include buoyant vehicles (balloons, airships) and heavier-than-air aircraft, specifically fixed-wing airplanes, both fueled and solar-powered. In all cases, the extremely low air density has dramatic effects on the architecture and sizing of the vehicle and so also the cost.

The low density means exceptionally large balloons are needed to displace enough air mass to compensate for the mass lifted for buoyant vehicles. This increases the cost of the balloon and the cost and complexity of its operations, including launch and recovery.

For fixed-wing vehicles (airplanes), the low air density means that wings cannot generate as much lift per area, so the vehicle needs to go faster and/or the wings need a larger lifting surface area. The problem with both means significantly more energy expended vs. a lower-altitude airplane carrying the same payload. A solar-powered airplane requires much more solar area, and much more battery mass, both of which require the vehicle to be larger still, with a corresponding increase in cost. Likewise, the large wing area required for lift means that the structural weight necessary for the wing plus the batteries uses nearly all the lifting capability, leaving little to no weight budget for a payload. There is no battery weight for a fueled airplane, but these factors mean that more fuel is required, or the flight time is shorter.

A significant disadvantage of a fixed-wing aircraft versus a buoyant aircraft is that propulsion power is required all the time. For solar-powered fixed-wings, that means batteries are needed to supply that power all night long, making overnight flights challenging with existing battery technology. Such vehicles often have record-breaking wingspans with extremely high aspect ratio (very long and skinny) wings, required for maximum efficiency. In all such aircraft, because their energy and weight budgets are operating so close to the limits of current technology, the incremental cost of larger payloads is much higher than it is for fueled fixed-wing and buoyant vehicles.

One way to address the serious energy budget problem for a solar-powered stratospheric fixed-wing aircraft is to drive the aircraft with fuel. All of the existing stratospheric airplanes, including the SR-71[11] and the U-2[12], reside in this camp. It also includes several proposals for Hydrogen-powered airplanes.[13] The advantage of hydrogen is that it has the lowest weight for any energy source except nuclear. The disadvantage is that it is a gas and retaining that gas in a liquified (super-cooled) or compressed form adds significant weight to the vehicle for the vessel. Nonetheless, several analyses show that a slow (not supersonic) liquid-hydrogen-powered airplane could carry a decent service payload and reach flight times of a week or more.

One of Loon's original assumptions was that relatively simple and inexpensive balloons with flight times of months could provide more cost-effective services than very large, very efficient, but very expensive, solar-powered aircraft with similar service timespans. Loon learned that it wasn’t that simple.

A potential middle ground would be a stratospheric airship that takes advantage of the buoyancy to lift a sizable payload capable of delivering a high-value service. This airship uses lateral propulsion (propellers and motors) and an aerodynamic shape to remain over the service area without frequently being blown away. Combined with Loon's altitude-based steering systems and advanced navigation algorithms, the propulsion means customers can count on the service being available when needed and that the vehicle is serving and gathering revenue for most of its flight time. Loon made serious headway towards designing such a vehicle, showing that this is a very promising approach.

11 https://en.wikipedia.org/wiki/Lockheed_SR-71_Blackbird
12 https://en.wikipedia.org/wiki/Lockheed_U-2
13 https://www.stratosphericplatforms.com/

Loon's Encounters with Stratospheric Weather

Operating in the stratosphere presents some difficulties due to the low air density and tough thermal environment. Still, the way Loon used winds in the stratosphere presented even more difficulties and more learning opportunities. Loon's use of different wind directions at different altitudes meant that balloons would get blown away from the service area when the winds were all heading in the same direction. Periods of time with such aligned winds were trouble. If they lasted more than a day or two, it was challenging for a balloon to stay in or make its way back to that service area.

QBO and Other Stratospheric Cycles

One particular pattern found in lower stratospheric weather was the Quasi-Biennial Oscillation (or QBO). The QBO is a roughly two-year cycle in the tropics where the winds in the stratosphere change from almost all eastward to almost all westward and then, a few months later, change back to eastward. These changes start from upper altitudes then make their way down lower, taking between 6 months and more than a year to complete each transaction.

Figure 1-9 is a wind-direction chart of the stratosphere that shows the wind direction shifts. *Red* means eastward, and *blue* means westward.

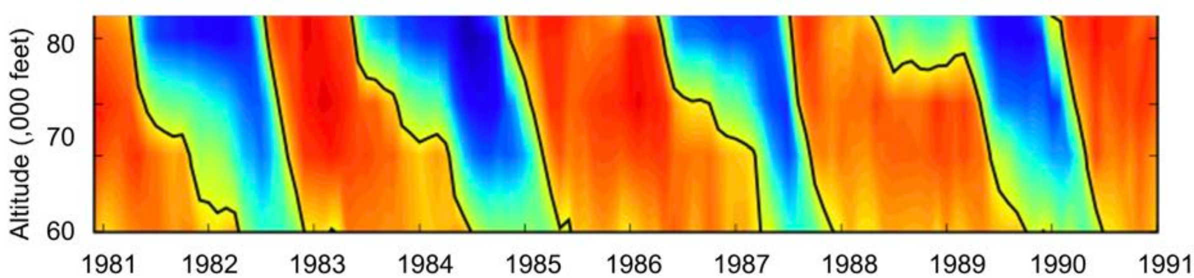


Figure 1-9 ***Wind direction chart of the stratosphere 1981 – 1991.***
You can see a relatively clean cyclical pattern with the winds changing direction from the higher altitudes down to the lower ones with an occasional deviation from that pattern. Where this is trouble for Loon is when the whole altitude range is the same color, such as in late 1985 or at the end of the chart, approaching 1991. These periods would blow the fleet away from their targets, requiring the balloons to take one to several weeks to return to the service area.

Beyond the QBO, several other known phenomena impacted Loon:

- The MJO (Madden-Julian Oscillation) shows some correlation with poor steering in Peru and over the Pacific Ocean. However, using forecasts of MJO didn't provide Loon with any more bad-steering forecast accuracy than simply using ECMWF HRES forecasts.
- NAO (North Atlantic Oscillation), north polar vortex, and turnaround are the three significant patterns that go far to explain Loon's navigation flying away from our launch facilities in Winnemucca and Puerto Rico. Different phases of these patterns could be correlated to launched vehicles heading toward to Canada vs. Europe vs. over the Pacific.
- When flying in the southern hemisphere, Loon also had to watch for the south polar vortex breakage. It would jump north out of Antarctica and swallow Loon's balloons and pull them farther south. A similar effect occurred in the northern hemisphere, though less commonly.
- The southern hemisphere turnaround effect (along with QBO, described above) did degrade Loon's steering across much of the targeted latitude band because of the descending easterly winds and because those winds made the balloons circumnavigate the world in the other direction.

These weather cycles and patterns are known to the stratospheric scientific community, but it was up to Loon to understand how they impacted Loon's operations and service quality. After analysis and real-world encounters, Loon developed tools and algorithms to better predict and deal with QBO. The only good thing about the pattern is its seeming predictability, which allowed us to forecast good and bad steering seasons. Unfortunately, the last few cycles have sputtered and back-tracked, with these anomalies being worse than ever seen since these measurements were started, potentially decreasing the predictability but also potentially changing (for better? for worse?) the impact of QBO on wind-based navigation.

There's Weather Up Here

Although weather is not as extreme as it is in lower altitudes, Loon's stratospheric vehicles are still subject to turbulence and electrical activity due to tropospheric storm activity well below. Stratospheric vehicles need to be robust against these effects. More on these effects and their impact on Loon's operations and engineering are described in "Flight Vehicle Role and Challenges" on page 51.

Loon's operations over South America provided the first demonstration that the vehicles weren't completely free of turbulence at 65,000 feet, even in clear weather. In December of 2016, there were several anomalies as balloons were transiting over the Andes. Analysis of forecast data and satellite imagery showed large waves of air moving up and down, called gravity waves, generated from the jet stream flowing over the Andes. These waves reached well up into the stratosphere, causing turbulence at Loon's altitudes. Balloon telemetry suggested that the balloons encountered these waves, which caused sufficiently heavy turbulence that the payload swung around far enough to hit the balloon envelope.

In addition to the actual weather in the stratosphere, another issue is the lack of visibility into that weather. There are very few direct wind or weather measurements made in the stratosphere compared to lower altitudes. Most commercial aircraft are equipped with instruments to measure and relay wind, turbulence, and temperature measurements and ground-level weather stations deployed extensively worldwide provide these and even more types of data. This wealth of data allows fairly accurate forecasts and calculations of past weather (analysis and reanalysis data). But, because the input data is so sparse for the stratosphere, forecast and reanalysis data quality is lower. Loon handled this by taking its own measurements; in addition to ascending and descending to take advantage of the winds, Loon's steering controllers also directed the balloons to occasionally visit altitudes to update Loon's data about the winds at those levels.

Despite these issues, there are good opportunities to utilize the stratosphere to provide various services, but the organizations doing so should keep these experiences in mind.

Chapter 2

System Engineering

The stratosphere presents opportunities for businesses and science but enormous challenges for the design and operation of a fleet of vehicles capable of providing viable services from the stratosphere.

Key Engineering Challenges

Several factors make working in the stratosphere difficult:

- The near vacuum (less than 1/20th as much air as at sea level) means that Loon balloons require at least 20 times the volume needed by a near-ground balloon to lift the same weight.
- The thermal environment presents challenges in keeping electronics and mechanisms warm when turned off and making it tough to cool electronics and mechanisms that generate heat. A mobile network service like Loon requires a lot of electronics.
- The operational life of most large production super-pressure balloons is a few weeks to a few months. Because lifetime of the vehicle is a critical cost factor, Loon needed to address and mitigate the film and seam leakage, thermal and UV stresses responsible for reducing the lifetimes. By 2021, Loon was nearing year-long flights and was working toward 400 days and longer.

- Loon's use of winds going in different directions for navigation required operation across a sizable altitude range to find and access that variability. It needed substantial solar and stored energy to push air into and out of the balloon for ballast and required balloons that were significantly larger and stronger than typical superpressure balloons because a high operating ceiling (maximum altitude) and a low operating floor were necessary.

The primary challenge for Loon was dealing with these issues while meeting the need for scalable and economic operations:

- The overall cost of providing service had to remain low to offer an attractive and profitable service. Maintaining service at a low cost was needed for markets beyond Loon's target focus and services besides mobile network expansion.
- Loon targeted large operations that would require thousands to many thousands of balloon launches and landings every year. This scale affected many elements of the design and operations, to reliably produce, launch and recover so many balloons so quickly.
- Operation of such a large fleet of vehicles still required low operating expenses on a per vehicle flight hour basis, so the number of human operators had to scale much more slowly than the number of balloons in flight. Loon operations required a great deal of reliable automation, capable of independently handling all but very exceptional events.

Loon Technical Overview

The Loon system was large and complex and extended across the globe, encompassing more than just a fleet of airborne flight vehicles. It also included launch facilities, ground stations, ground networks, satellite communications, a large amount of software running in Google's data centers, and many other supporting elements.

The airborne flight vehicle itself is more than just the balloon, which itself is much more than just a bag of helium. The balloon consists of:

- An inner ballonet and an outer envelope plus tendons holding the envelope together and load rings holding the tendons
- The air pump (ACS, or Altitude Control System) that controls the altitude of the balloon

- The electronics and mechanisms at the top and bottom of the envelope that:
  - » Enable fill and attachment during the launch process
  - » Perform flight termination
  - » Perform pressure and temperature monitoring of the lift gas (in the ballonet) and the air ballast (in the outer envelope)

The rest of the vehicle, which the balloon carries, hangs on a structure below and includes:

- Solar panels, battery systems, power distribution, and control systems
- Complex electronics, including more than a dozen computers of various types and many electronics boards
- Two satellite communication systems for redundant control and telemetry
- Multiple GPS systems for position and heading determination
- A service payload, including a full multi-sector LTE base station, three advanced 1 Gbps gimballed radios, and multiple on-board networks (Ethernet, Controller Area Network or CAN)
- A variety of advanced thermal control mechanisms
- The Automatic Dependent Surveillance-Broadcast (ADS-B) transponder to communicate with air traffic control and other aircraft
- A solid powder ballast release mechanism
- A parachute to land the system safely
- A considerable amount of software that controls each of the above systems, individually and as a working whole
- Plus, more supporting structures, mechanisms, and electronics.

In addition to the flight vehicle, on the ground, there are:

- Auto-Launch cranes and the Final Assembly and Hatchery facilities at multiple Launch sites
- Ground stations for providing internet backhaul to the balloons, which are located at launch sites, test sites, and across service areas
- Network routing, management, and controlling systems, found in data centers or mobile network partner's network centers, connected via fiber optic cables to the ground stations and the partner's centralized networking systems

- Two Loon operations centers, where flight engineers and other operations personnel meet, monitor, and discuss the operation of the fleet and communications services
- And in the cloud (in Google data centers around the world), all the software required to operate the entire fleet of Loon vehicles safely, responsibly, and efficiently

Loon in a Nutshell: the End-to-End Operation

This whole system works together in the following way:

Assembly and Launch

- The numerous advanced components making up the Loon vehicles are built at various contract manufacturing sites worldwide and shipped to the launch facility. There, final assembly, test, software installation, and configuration occur before wheeling the vehicle to the Auto-Launch rig for a largely automated launch.
- The balloons appear largely empty at launch, with what looks like a small bubble of lift gas (helium) at the top of a big floppy plastic bag. This small bubble is sufficient to lift the balloon and its payload off the ground and have it ascend quickly. During this ascent, the small bubble of helium grows as the surrounding air pressure diminishes with increasing altitude. Eventually, the bubble fills the entire balloon when it reaches the float altitude. At this point, the envelope is no longer floppy but is taut, forming a solid pumpkin-shaped balloon.

Altitude Control for Steering and Navigating

- Because the outer skin of this balloon is not stretchable (unlike a toy or party balloon), air pumped into the balloon by the Altitude Control System (ACS) increases not just the overall weight but also the mass density, causing it to sink in the atmosphere. The opposite operation makes the balloon ascend: pumping the air back out of the balloon reduces its weight. The injected air and the lift gas (helium) are in two separated compartments (outer envelope and inner ballonet, respectively) to keep them from mixing.
- Using this ascent/descent control, Loon's software directs each vehicle up and down to find and take advantage of winds going in different directions to navigate the vehicle from the launch site to the designated service area.

Connecting to the Internet and Connecting the Users

- When the vehicle arrives at the service area, Loon's network orchestration software instructs one or more of the vehicle's three gimballed backhaul antennas to point to a ground station and/or other Loon vehicles already in the area. In this way, the balloons in the service area form a balloon mesh network connected to the ground networks via one or more ground stations, which are connected via fiber to the Loon and mobile network operator's network there, to the internet. The in-air mesh network, which often reaches thousands of kilometers, simultaneously connects dozens to hundreds of balloons to the ground network through a single ground station or several redundant ground stations.
- With that backhaul and internet connection in place, the network and LTE systems on the vehicle can turn on and direct the LTE signal to the ground, where users can connect with standard LTE phones. From the user's perspective, there is no indication that their connection is via a stratospheric balloon, other than that they're standing in the Amazon or rural Kenya, locations where users had never had mobile phone or internet service previously.
- As winds change and vehicles rearrange their locations by ascending and descending to keep the vehicles in or near the service area, the mesh network continuously adapts, removing old and creating new links between vehicles and between ground stations and vehicles, actively creating the most resilient interconnections. Loon maintains the backhaul to the LTE systems on all the vehicles using this methodology.

Pushed Away and Flying Back

- Occasionally, across the entire altitude range reachable by the vehicles, the winds are all going in roughly the same direction, which forces vehicles to drift away from the service area. A second layer of vehicle navigation algorithms kicks in at that point, plotting a path around the region to return to the service area.
- If these poor wind periods continue, the Loon navigation system then orchestrates and schedules the arrival and pass-through of each vehicle, one after the other, to provide continuous service as each vehicle follows the last.

Approaching the End of Life

- The end of the vehicle's life is determined either by remaining helium quantities or by the natural degradation of the vehicle's components due to prolonged exposure to the harsh environment. Before reaching this point, the flight engineers command the system to navigate the vehicle to a landing zone.
- The fleet management software helps the flight engineers find an optimal location to start the descent to maximize the chances of landing within ten km of the target. As the vehicle approaches the landing zone, the software continuously predicts the landing location based on the expected descent rate and the expectation for the direction and speed of the wind layers along the descent path. The navigation systems then choose an optimal route to the location chosen.

Descending, Landing, and Recovery

- After reaching the recommended start-of-descent coordinates, the flight engineers command the flight termination system at the top of the balloon to cut holes in the balloon and ballonet, releasing the lift gas. As the vehicle descends, it begins to pick up speed, and eventually, a landing parachute is released, reducing the vehicle's speed for the last portion of the descent.
- Loon dispatches the local vehicle Recovery Team to where the vehicle is expected to land. The Recovery Team tracks the descent and after landing, loads the balloon and payload into a truck and drives it to a facility for disassembly for recycling, reuse, or subsequent engineering analysis.

The following are some terms relevant to the subsequent sections:

- ***Flight vehicle***: Refers to the entire flight system, including the balloon, bus, and payload.
- ***Balloon***: Refers specifically to the lifting body (envelope, ballonet, apex and bottom assemblies, and ACS). But, in some areas, when consistently referring to the entire airborne system, "balloon", "flight vehicle" and "vehicle" are used interchangeably.
- ***Weight vs. mass***: Although weight and mass are slightly different concepts, they are used somewhat interchangeably in this document, but we only use kilograms (mass), not newtons (weight: $1 \text{ N} = 1 \text{ kg·m/s}^2$).

- ***Air and gas pressures***: Described in pascals (1 Pa = 1 N/m^2), which also denote altitudes, a common meteorological convention for stratospheric discussions. Air pressure decreases with altitude, so higher Pa numbers indicate lower altitudes. A helpful rough guide:
    - Sea level is about 100,000 Pa (actually, 101,325 Pa). Where Loon operated, the stratosphere starts somewhere between 20,000 Pa (lowest altitude) and 10,000 Pa (highest altitude), which means the air pressure at that point is about 1/5th to 1/10th that of sea-level air pressure.
    - The bottom of Loon's typical range for balloons over the past few years was about 10,000 to 11,000 Pa. (About 1/10th sea level air pressure.)
    - The top of Loon's typical range was 5,000 to 6,000 Pa. (About 1/20th sea level air pressure, or 1/2 Loon's operating floor.)

Loon's Superpressure Balloons–Architecture and Operation

The balloon shown in Figure 2-1 is the most visible and unique part of the flight vehicle and consists of several main elements:

- The clear plastic outer *envelope*, which holds the air ballast
- The internal plastic *ballonet*, which holds the lift gas (helium)
- The *apex* and *bottom assemblies*, which contain a variety of electronics, sensors and mechanisms, the altitude control system, and the load rings, securely holding the envelope tendons (which are under enormous strain due to the overall pressure in the balloon)

At the *float* (operational) altitude, the balloon reaches its superpressure state. Its internal pressure is greater than the surrounding air, causing that envelope to expand to its full, pressurized, and taut pumpkin shape. The superpressure is a measure of the difference in pressure between the inside and outside of the balloon, so a positive superpressure means that the balloon is taut. In contrast, a zero superpressure means the balloon is no longer able to hold its shape.

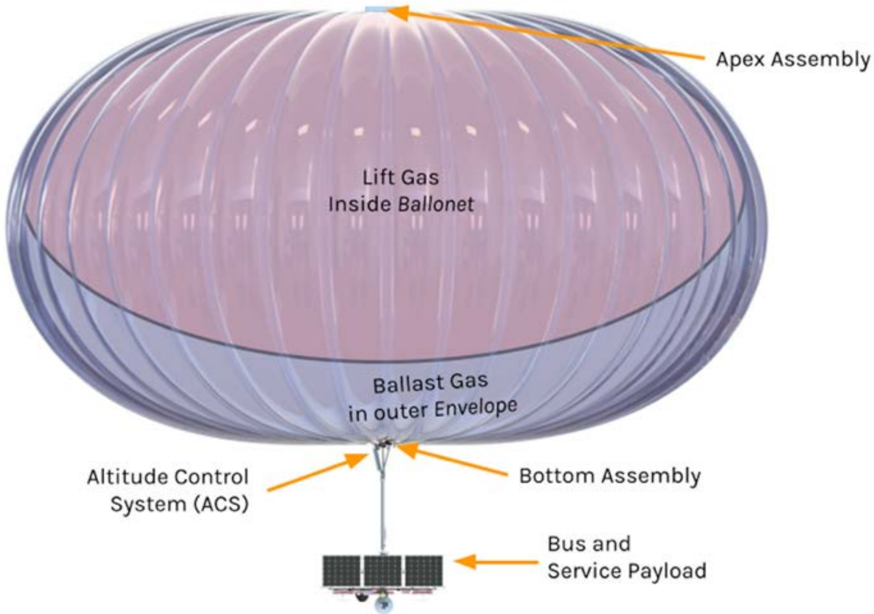


Figure 2-1 Superpressure balloon overview.

Within the outer envelope, the inner ballonet holds the helium lift gas. If this gas were instead in the outer envelope, as it is with almost all other superpressure balloons, the pressure difference across the outer envelope film would continuously push the helium to escape through that film. Having the helium in the ballonet solves this problem. The ballonet film has helium gas on the inside and ballast gas outside at the same pressure, which significantly reduces helium leakage. In turn, the balloon lasts significantly longer. This is one of many advancements made by Loon to decrease the cost of operations.

The air in the outer envelope, called *air ballast*, is blown into the balloon using a pump called the Altitude Control System (ACS). This blown gas, or air ballast, adds weight to the system. Therefore, removing the air ballast (pumping it back out again) has the same effect as emptying ballast bags of sand overboard on a hot air balloon: it causes the balloon to ascend. This gas-in => balloon-descends relationship works because the outer envelope is nearly inelastic, so the overall volume of the balloon changes very little, causing the overall density of the balloon to increase. If the air ballast is pumped out, the ballonet can expand to nearly the full volume of the outer envelope, which will be the case when the balloon is at its maximum operational altitude.

The physics of the float, ascend, and descend processes is more complicated than it first appears due to the thermal and pressure environment outside and inside the balloon:

- As the balloon descends or ascends, the external air pressure increases or decreases
- The internal gas temperature is affected by several factors:
  - » The gas warms slightly as it is compressed in the balloon or cools slightly as it expands during the release of the air
  - » The temperature of the ambient air does slightly change across Loon's altitude range
  - » The sun's direct radiation and the reflected radiation from the earth, the clouds, and the infrared energy radiated by the earth at night all affect the temperature of the lift and air ballast in the balloon

The ascent or descent does stabilize, although sometimes with a small up/down cycle as these factors take a while to settle out.

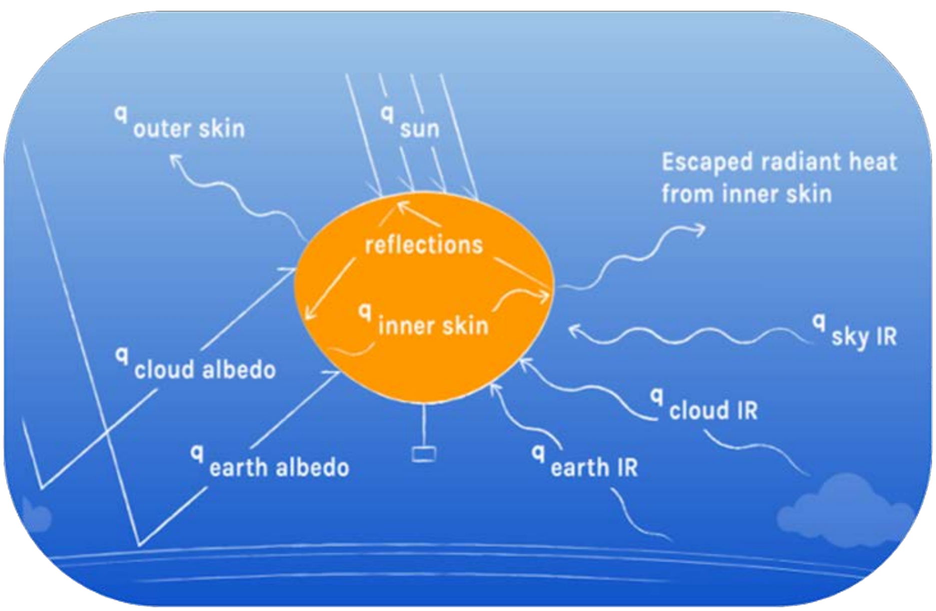


Figure 2-2 ***Physics of balloon thermal environment.***
The temperature of the internal gases are subject to both conducted and radiated thermal effects.

Over the life of a vehicle, the helium will slowly leak through slight imperfections or holes or even slowly permeate the ballonet material itself. As this happens, the maximum altitude of the vehicle will get lower. Eventually, this loss of lift gas will keep the vehicle from remaining at the desired altitudes, marking the end of the vehicle's service life.

Key System-level Metrics, Objectives, and Constraints

At the highest level, Loon's key challenge was not just technical but economic. To achieve the mission of providing access for people in far rural areas, we needed to do so with a system that could scale economically. The overall cost of providing that service had to remain low to offer an attractive service for those users we wished to connect, and it had to remain low, even when serving millions of users across large portions of Africa, South America, and Asia.

For this reason, the key metrics for Loon were the same as for most other businesses: gross profit and customer-perceived product value (e.g., capabilities, quality, and performance). In Loon's case, the main product was the mobile network expansion service, and our direct customer was the *mobile network operator* (MNO) acting as our partner. The MNO's customers were the mobile network end-users and, because the partner's success depended on their users being satisfied with the service, Loon also worked to meet the mobile users' needs.

Gross profit is simply the money earned from the service (the revenue) minus the cost of providing that service:

Potential ***revenue*** structures vary across customers or services, but a typical approach for such services is a revenue share, a fraction of the revenue that the operator earned from the data that Loon served. As with many mobile network services, this revenue most highly correlates to the amount of data delivered to users (measured in gigabytes, GB). Therefore, GB served is the best metric for Loon to use to judge the overall system's revenue capability.

- The ***cost*** of providing Loon's service includes the costs of the vehicles, the ground infrastructure, and the cost of all other Loon operations related to providing the service. The most significant components of the costs of providing Loon service are the vehicle and vehicle-related costs. These also include the personnel and equipment costs of manufacturing, launching, operating, and recovering the vehicles.

To scale economically, Loon then needed to focus engineering efforts on lowering the cost per GB served, with the target being lower than the revenue per GB served. Due to the sensitivity to winds and weather, this metric was calculated and optimized for the whole fleet and over time.

Potential revenue structures vary across customers or services, but a typical approach for such services is a revenue share, a fraction of the revenue that the operator earned from the data that Loon served. As for most mobile network services, this revenue most highly correlates to the amount of data delivered to users (measured in gigabytes, GB). Therefore, GB served is the best metric for Loon to use to judge the overall system's revenue capability.

The ***perceived service value*** for a mobile service mainly depends on two key service performance metrics:

- ***Data rate*** or throughput, typically measured in megabits per second (Mbps)
- ***Availability*** (does it work where and when you want it to), typically measured as a percent of time service works, e.g., 98%

It was important to Loon that we provide a good service both to keep the users satisfied as well as to support our partner operator. If the users experience slow data throughput and service outages, the experience reduces the amount that users are willing to pay. It may also give customers a negative view of the operator, affecting the operator's adoption and user satisfaction numbers outside of the Loon operations. Likewise, if the service is not available when the user expects it to be, that also reduces the value of the service to the user.

For communications services, availability is defined in several ways. It is generally considered to be the amount of time that service is working divided by the amount of time that is being tested, with this test period typically being a month. If more consistent service is valued, the test period should be shorter: a week or even a day.

In addition to the revenue-related metric of the number of GBs served, Loon needed to track a variety of coverage metrics to understand the service quality and characteristics that we were delivering. For instance:

- Number of unique users served
- Average number of users per day
- Amount of land covered
- Serving time over target

These metrics were valuable to show that Loon was reaching areas that had not been reached before. In situations where Loon could receive a coverage premium, then one or more of these coverage metrics could be integrated into the revenue equation. For most Loon contracts, however, the revenue was still largely dependent on GBs served.

Key Learning: Loon Engineering's Primary Objectives

Summarizing the above, Loon's engineering task then was to design a system that:

1. Minimized the cost per GB served to reach economic scalability,
2. Met the required service levels so users perceived a high value to the service,
3. Operated safely and reliably in the stratosphere,
4. And was realizable, that is, could actually be designed and built.

As we show in the following sections, the first two were very interdependent; improving service quality will almost always come with a cost and reducing service levels may reduce the revenue that can be earned. The last two items were absolute design constraints, though there were many different ways to meet those constraints.

Design Considerations: Performance, Complexity, and Cost

To improve the service level or quality, a service like Loon had three main levers to move:

- Improve the performance of ***each vehicle***, or
- Increase the ***number of vehicles*** in the fleet, or
- Improve the way the vehicles ***operate as a fleet***, and this one generally does not increase costs

Improving the performance of each vehicle or increasing the number of vehicles in the fleet are choices that can increase availability and data throughput but do, in most cases, increase costs. Improving the software automation and algorithms to operate the fleet more efficiently only affects engineering costs, not per-vehicle or per-GB costs and so Loon invested heavily in this area. These efforts are detailed in this chapter: "Loon Software" on page 234.

For a new vehicle design, many design variables could be considered, with a tradeoff matrix so complex that a Model-Based Systems Engineering method is required to determine and have confidence in the design choices. With such an approach, detailed and accurate models of the system and environment are used to understand the impact to costs, performance and revenue, of many proposed configurations. Loon developed and used such tools to analyze millions of potential vehicle designs, working towards a set of optimal designs.

Fleet Size and Overprovisioning

Were balloons static, constantly hovering over their service area, Loon would need only a small, fixed, and predictable number of vehicles to provide coverage for each patch of land. Service would be available at nearly constant levels. But balloons are not static; they float along with the wind, often away from the coverage area, until the algorithms find the right winds to sweep the balloons back to the coverage area, days or weeks later. Even when vehicles are near the target, the flight paths often overlap or leave gaps in the coverage.

Loon kept service availability from dropping below acceptable levels by dedicating extra vehicles to each service area so that when some drifted away or were not positioned optimally, others could fill in and keep the service operating. The ratio of vehicles dedicated to a service area versus the minimum number of fixed location vehicles is called the ***overprovisioning*** (OP) factor. Let's look at an example:

Assuming a single vehicle's potential service footprint is roughly 5,000 km^2, a particular service area about four times that size could be served by four vehicles if those vehicles were fixed to a chosen location. However, depending on the winds, those four vehicles would drift around the area and occasionally be blown far away, requiring a few days or longer to return. To solve this, 20, 40, or even more vehicles may need to be dedicated to serving that area to have decent confidence (e.g., 75% likelihood) of hitting a chosen availability target (e.g., 80%). Even more vehicles would need to be flying and available for even higher confidence or higher targeted availability.

In this example, if 40 vehicles are required to meet the availability objective at the confidence necessary, then that would be an overprovisioning of 10x. For this same location and vehicle design, overprovisioning could change to 5x by reducing the availability requirement or 20x if higher confidence is required.

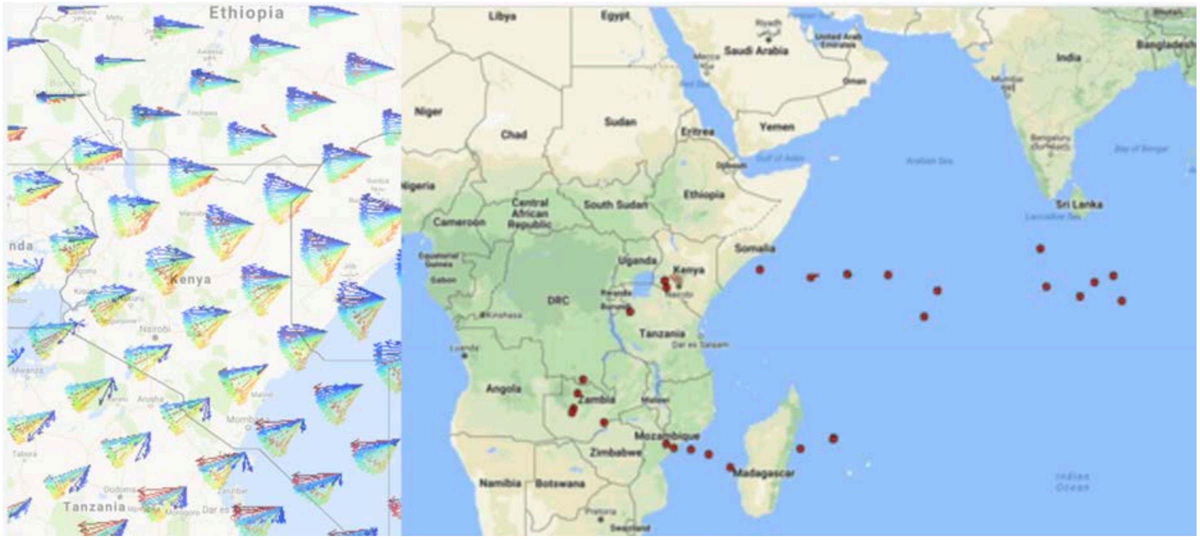


Figure 2-3 ***Wind cones and overprovisioning.***
Simulation of overprovisioned fleet serving southwest Kenya, with a 10,000 to 15,000 km return loop. On the left are the different wind directions found across the balloon's altitude range showing that, across Kenya, vehicles will be consistently blown westward.

Figure 2-3 shows another example of overprovisioning in a simulation of v1.4 balloons serving southern Kenya. In this case, the service area covers about 6 balloon footprints, but 30 balloons are allocated, for an overprovisioining of 5x. The counter-clockwise circuit of balloons looping through east Africa then through the Indian Ocean is an emergent pattern created by Loon's steering algorithms. The wind cones shown to the left of the map show that all the winds accessible to balloons over Kenya will force the balloons towards the west but after that, they can then make their way south and eventually back out east where they can head north and then west to fly again over Kenya. This 5x overprovisioning does not ensure perfect availability but in combination with Loon's advanced navigation algorithms, Loon was able to provide decent service even in these challenging wind conditions.

Clearly, high overprovisioning factors increase the Loon cost of operation. But if each of those additional vehicles could deliver the same number of GBs as those in a smaller fleet, it would not affect the overall cost per GB. This shows that the important factor for understanding the cost per GB impact is not overprovisioning but fleet utilization: the percent of time the vehicles in the fleet were in a service area and could earn revenue. Doubling the size of a service area might barely affect the overprovisioning factor, but could cause the fleet utilization to go from 5% to 10%. If Loon could serve users anywhere in Africa, the overprovisioning would still compel huge fleet sizes but the fleet utilization will be much higher, potentially 90%

or more, as those vehicles would spend almost their entire lives over the continent, serving users.

Overprovisioning versus Availability versus Confidence

Loon had downloaded decades of historical weather data that we used to estimate the quality of the service as it was deployed in new areas or with new vehicle designs. By simulating fleet behavior across those decades, Loon could determine, for a particular targeted service area and a specific vehicle design, how often the service would have met a particular availability level with different fleet sizes. For example, perhaps half of the tested months hit 90% availability for a given fleet of vehicles, but three-fourths hit 70% availability. From this, it can be extrapolated that roughly the same ratios will happen over time, which could give a rough confidence level for a targeted availability and fleet size. Due to the strong seasonal effects on the winds, this analysis could also be sliced by season or month of the year (e.g., all the March months over the decades), or by another important weather phenomenon, such as the QBO, described in "Loon's Encounters with Stratospheric Weather" on page 21. This type of analysis was used by Loon to judge potential service levels in different service areas and estimate fleet sizes required.

Balloon Characteristics and Impact on Cost and Service Performance

The design factors on a vehicle that can improve steering are several:

- Increase the operating ceiling or lower the operating floor to find and use more wind variations.
- Increase the speed of ascent and descent: Takes less time to change direction and reduces the station-keeping figure-8 area. For more information, see "Navigating on the Wind" on page 255.
- Have more energy available for more ascending and descending per day: More maneuvers reach more favorable winds more of the time.
- Add a propeller for some lateral propulsion.

The v1.6 vehicle shown above had all improvements:

- The altitude range was increased, and the altitude ceiling was raised
- Two ACS units were used, which allowed faster ascents and descents, with little increase to the energy consumed per km of descent (moved the same amount of air twice as fast)

- The angle of the solar panels was slightly adjusted, increasing the solar power harvest
- Lateral propulsion was added in the form of a 2 m-diameter propeller Loon named Seahorse

These factors helped contribute to the improved steering performance of v1.6, but unfortunately, Loon only had a chance to fly a few v1.6 flights. The lateral propulsion was only capable of a sustained rate of less than 1 m/s, which meant that the vehicle is still largely being blown by the winds. However, simulation and analysis showed that this 1 m/s still had a noticeable effect on availability, overprovisioning factor, and fleet utilization. As expected, the improvement was very dependent on region and season; in some cases, the propulsion helped a lot and, in other cases, very little.

The next vehicle planned from Loon would leverage such gains even more by replacing the pumpkin-shaped envelope, which takes a lot of power to drag laterally through the atmosphere, with a streamlined, aerodynamic shell. With additional solar panels and batteries, Loon could drive that lateral speed significantly higher. These enhancements are discussed in "Transitioning to Stratospheric Airships" on page 351.

Better Altitude Range Improves Steering

Since Loon takes advantage of different wind directions at different altitudes, we can attain better steering and increased ability to stay near a target by going higher (extending the operating ceiling) and lower (extending the operating floor). Going higher requires a larger balloon due to reduced air density. Going lower (or increasing the pressure delta range) requires adding even more air into the balloon, which increases the pressure inside the balloon. This increases the stress on the PE plastic sheet, the tendons, and the load rings, requiring stronger versions, which increases the weight of the balloon, requiring it to grow larger as well. This feedback loop doesn't make it impossible to extend the maneuvering range; it is just more difficult and more expensive than it first appears.

Increasing the Power Available or Using Power More Efficiently

For every element of the design, low weight and low average power are two key goals. The vehicle's electronics and mechanisms require power; all that power is harvested with solar panels and stored in batteries for operation at night. The more power used when the subsystems are active or, the longer those subsystems are active each day or night, the more solar and batteries need to be added, adding cost and weight. This is particularly true of the ACS and LTE subsystems.

If more energy is available, then more altitude maneuvering is possible and, if lateral propulsion is present, then higher speeds could be reached.

Decreasing Weight of Components

The cost-performance relationship of the vehicle is not simple. For example, although components of lighter weight (but equal performance) generally cost more, they can more than earn their keep. It may cost more to use a carbon-fiber structure instead of a heavier aluminum one. Still, if it weighs a few kilograms less, it may pay for itself using that saved mass for more batteries or solar panels, which can improve steering, which can increase revenue. Or it could decrease the size of the envelope, which could lower costs. Another example: though supporting a wider range of altitudes may cost more, that will improve steering, so fewer balloons may be required, and so reduce overall costs.

Service Payload Characteristics and Impact on Cost and Service Performance

Loon's mobile network expansion service was delivered using a fully functional and compliant LTE base station with capabilities like cell towers found along highways. Loon operated most of its service with a four-sector base station, while typical terrestrial cell towers have a three-sector arrangement. Each of these sectors has separate antennas pointing to a different plot of land. Many of the performance constraints of LTE are based on per-sector limitations, so, with more sectors, Loon could serve more users, and those users could consume more data, generating more revenue.

The three most important factors to consider when estimating data throughput and the number of users for a particular payload design and configuration are:

- ***Sector count:*** As described above, roughly, the LTE capacity of a particular tower (or balloon) is proportional to the number of sectors.
- ***Sector bandwidth:*** Most LTE systems are 10, 15, or 20 MHz bandwidth, and the capacity is roughly proportional to those values. Loon's later LTE systems were fully capable of 20 MHz (the maximum allowed by LTE), but some markets or MNO partners were constrained to 10 or 15 MHz.
- ***Interference between sectors:*** If two sectors using the same radio channel are too close to each other, or their antennas point toward the same set of users, those sectors will have degraded performance because of the radio frequency (RF) interference between their signals. Essentially, the two sectors are talking over each other, and no one can hear as well.

The downside to adding more sectors is that they cost more in several ways: the cost of the electronics and antennas themselves is roughly proportional to the number of sectors, but so is the weight of the system and the power used by that system. That increased weight requires a larger balloon, more helium, and a larger launch crane: factors that drive the overall cost even higher. Likewise, the additional solar panels and batteries needed to support the increased power also increase the weight, again increasing the overall costs.

In addition to being a good proxy for revenue, the overall capacity of the service payload is important for another reason: a balloon in the stratosphere can service a huge footprint. That footprint can potentially cover many users, so the vehicle must have the capacity to provide adequate service to those users. In this way, the capacity impacts the revenue and can significantly impact the data throughput and availability of the service, affecting the end users' experience.

Other Factors with Impact on Cost and Service Performance

There are several key factors beyond those related to the balloon and payload design and the navigation algorithms that affect Loon's cost and performance:

- ***Seasonality and large weather cycles:*** Winds in the stratosphere vary unpredictably from week to week, but there are strong correlations with the season. Loon did not wish to only serve during some parts of the year, nor did partners want to see a significant reduction in service quality in some seasons. To enable year-round service, one option was to increase the fleet size during the bad wind periods and decrease it again when winds were good.
- ***Service location, size, and design:*** There are large differences in the winds from location to location, making some countries or regions more expensive to serve than others. Likewise, because the stratospheric winds in many locations tend to blow east or west more than north or south, the fleet would typically move mostly eastward or westward. Serving very wide (east/west) but short (north/south) service areas allowed the balloons to stay in the area longer than if serving tall, skinny areas, where the balloons will only make a brief appearance on their way through.
- ***Latitude range:*** Expanding the latitude range, that is, serving farther away from the equator, increases potential target markets that can be served, but does require more solar and batteries because of the longer nights and shorter days in the winter, but it also means shorter nights and longer days in the summer. Like the seasonal steering above, it is usually not an option to offer a communications service for summertime only. Still, if it is possible, it can reduce costs when operating at these latitudes.

- ***No-fly zones:*** To ensure that the balloons don't fly into areas that we have been asked to avoid, Loon's navigation systems usually add a broad buffer around no-fly zones. Placing a service area near a no-fly zone will seriously degrade the availability. Conversely, getting permission to overfly a country bordering on a service area could greatly affect the availability or required fleet size.
- ***Neighboring service areas:*** The fleet utilization factor can be significantly increased by covering more and more of a balloon's natural, wandering flight path with service areas where the vehicle is free to serve and earn revenue. For example, Loon's service in Kenya resulted in the balloons frequently flying over Uganda and the DRC due to the dominant east/west wind. If those were service areas, those travels would increase revenue with no impact to cost, or, more accurately, this would decrease the cost per GB served.

These factors, which have an impact on cost and revenue, were examined during all phases of the business development process, including finding candidate service regions, and discussing specific service areas, seasonal availability changes, and contract negotiations.

Systems Engineering Design Approach

The early years of Loon were experiments to understand what is possible or how Loon could make it possible. As Loon solved problem after problem, the team developed a deeper understanding of the real underlying engineering problems we had to solve and a much better appreciation of where we could make major versus minor improvements over how others solved these problems. We also began to deeply analyze how to improve service quality, decrease costs, and scale the system to the level required to attain our objective.

As Loon came to understand the critical design parameters and constraints, we needed to answer a key question:

> *How do we choose all the many interdependent design variables across the balloon and the payload to satisfy all the constraints, maximize the service quality, and maximize the profit of the overall system?*

The potential design space is so huge and the interdependencies so complex and confusing that the answer could only come from a model-based approach. Taking advantage of the learnings and innovations from the past years at Loon, we built a series of tools to simulate different subsets of the overall system at different levels of detail to answer different parts of this overall question.

Tools

Over the past couple of years, the following tools allowed us to work through a series of steps, from a potential design space of millions of vehicles down to just a handful of designs promising enough to warrant exhaustive analysis, and from those, to pick just a few specific design targets for our next round of vehicle development. The objective was to optimize the design and size of the vehicle to reach the lowest cost of revenue while hitting the required availability. This was only fully realized in the process to define the next major Loon vehicle, a stratospheric airship codenamed Hammerhead, described in detail in "Transitioning to Stratospheric Airships" on page 351.

Helium and Powder Ballast Fill–Monte-Carlo Analysis Tool

Used by Loon engineers and flight prep teams to understand, for a given vehicle design (including balloon size and attributes and payload weight), the optimum amount of lift gas and powder ballast to fill before launch, where the objective is to maximize the combination of balloon lifetime and steering ability (altitude range accessible) and to understand the expected lifetime of the balloon given that gas and powder ballast fill. See Figure 2-4.

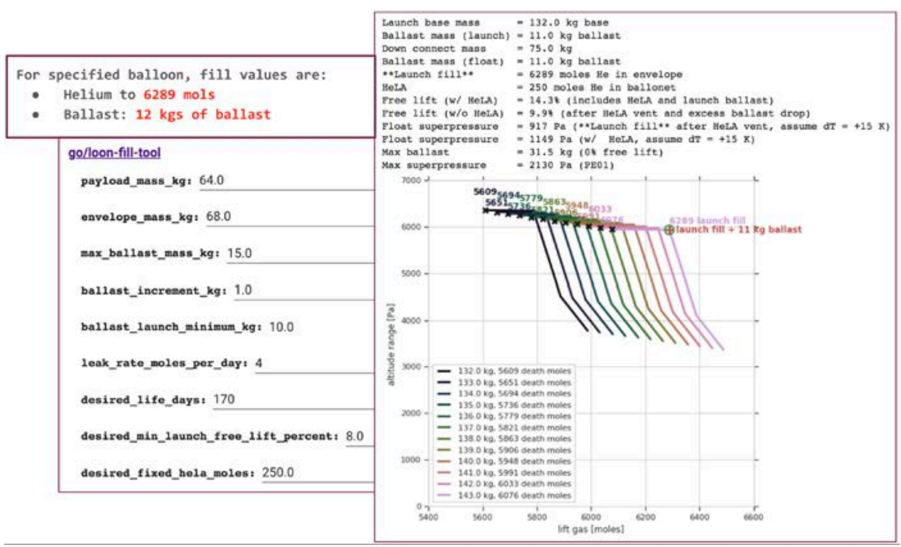


Figure 2-4 ***The fill calculator tool.***
The fill calculator calculates optimal helium and powder ballast to maximize steering range over longest flight time.

Power Monte-Carlo Tool

Used during new vehicle design to understand how often a specific vehicle design would be constrained on maneuvers or LTE serving hours due to exhausting stored energy before sunrise (dawn). Actual operations prevented this from happening by, if needed, turning off non-critical functions, such as LTE andACS, which would reduce the steering or communications performance of the vehicle. See Figure 2-5.

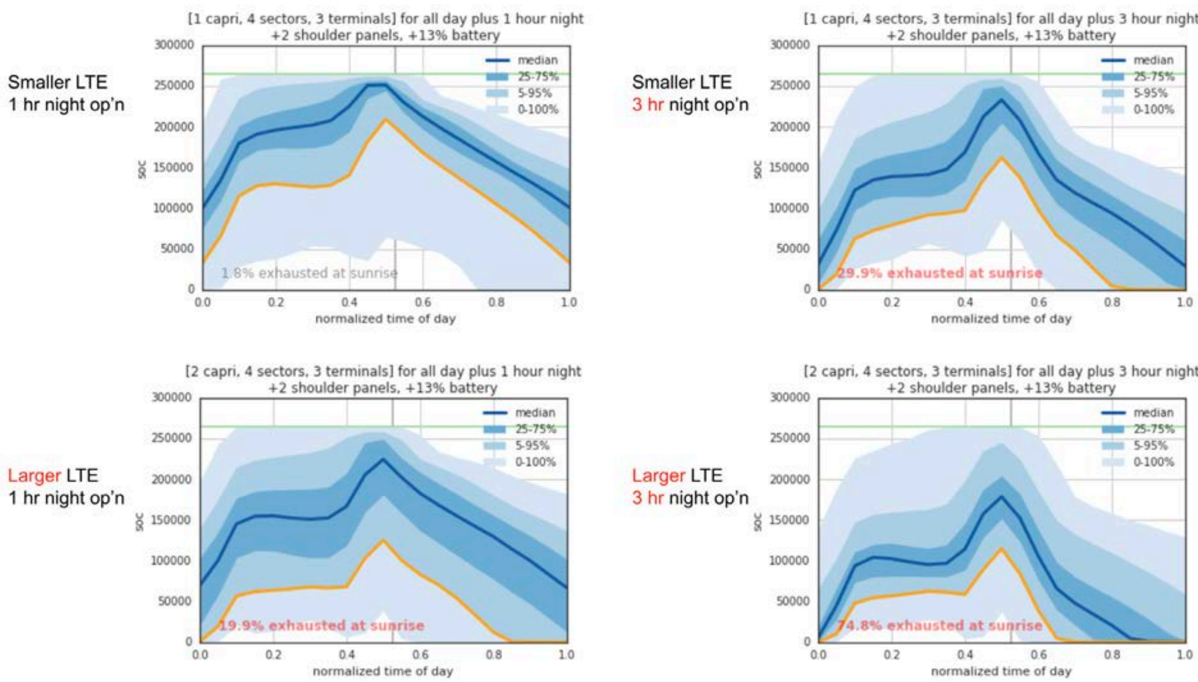


Figure 2-5 ***Monte Carlo analysis of two different payload designs.***
Two designs, operated for two different amounts at night: 1 hr vs. 3 hrs. Baseline (upper left) has lower capacity LTE and operates only 1 hr at night. Adding additional LTE capacity shifts the number of at-risk days from 1.8% to 19.9% but adding two more hours of nighttime operation increases the risk to 29.9% and 74.8% of days.

Vehicle Sizer/Optimizer Tool

Known simply as the Sizer, this tool was used to optimize a vehicle's design by tweaking many different sizing parameters towards a chosen objective metric (e.g., cost per flight-month). Sizing parameters might include envelope volume, battery storage capacity, number of solar panels, and size and number of ACS units. The tool supports a wide range of configurable constraints (such as payload weight and power, operating altitude range, and ACS descents per day). It checks all design parameters and resulting attributes of the optimized vehicle design to ensure a realizable design capable of meeting the given constraints.

The output of the Sizer was typically a CSV file containing one row for each of the viable candidate designs that met a chosen minimum performance, with each row including fields for all of the design values and calculated performance metrics.

Full System and Fleet Simulator: Very Small Fleet Experiment Infrastructure (VSFEI)

VSFEI was Loon's workhorse simulation tool. It incorporated a detailed model of the vehicle, including balloon, steering, and LTE payload characteristics. VSFEI accurately simulated the complicated interactions with wind and weather using the same steering, navigation, power control, and automation software to drive the production fleet. VSFEI could simulate the fleet over historical wind data going back decades, including the addition of wind noise, which was algorithmically defined to model the range of actual wind behaviors vs. the winds forecast for that period.

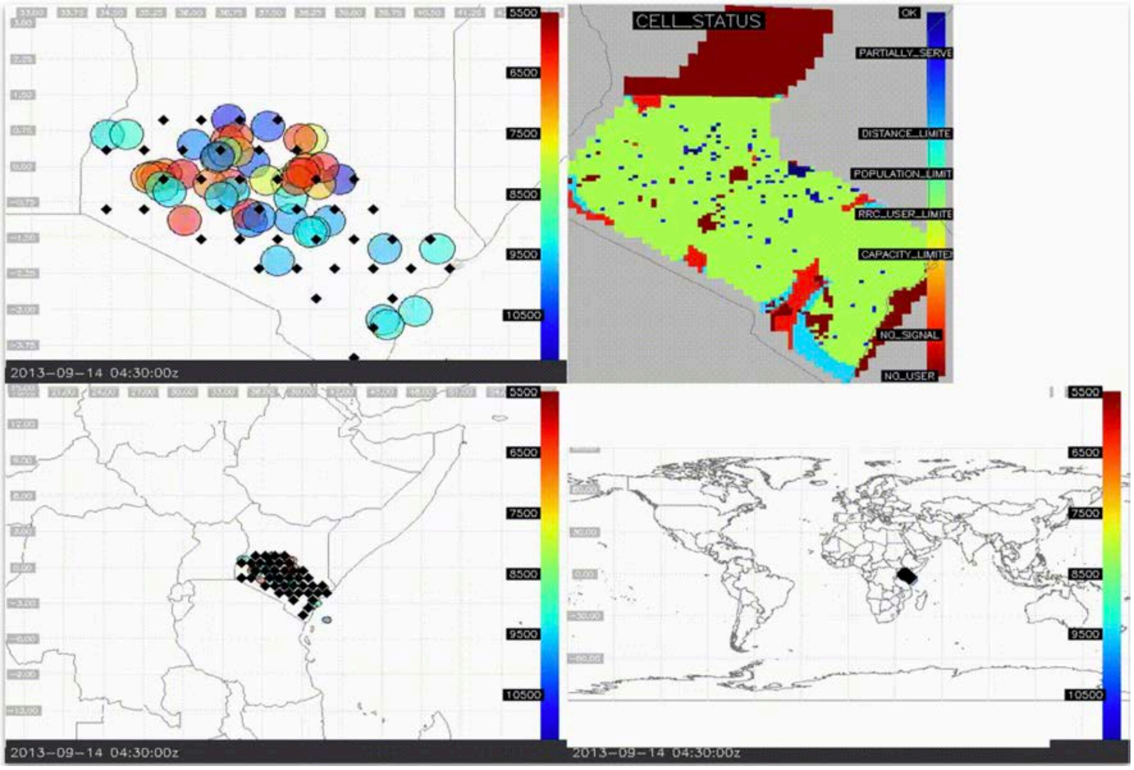


Figure 2-6 ***Loon fleet simulator output.***
Clockwise from upper left: Balloon LTE footprints and power status, overall LTE coverage status, global coverage, and regional coverage.

LTE Simulator

Loon's engineers used the LTE simulator to quantify the performance of a specific candidate LTE system on a fleet of balloons in a specific arrangement over a specific service area with a particular population density map and a specific adoption rate. This tool was used by Loon's production LTE control software (Airstream) to continually optimize the settings of the LTE sectors in the live fleet. It was also used in the VSFEI simulator (above) and other tools for engineering payload analysis and optimization.

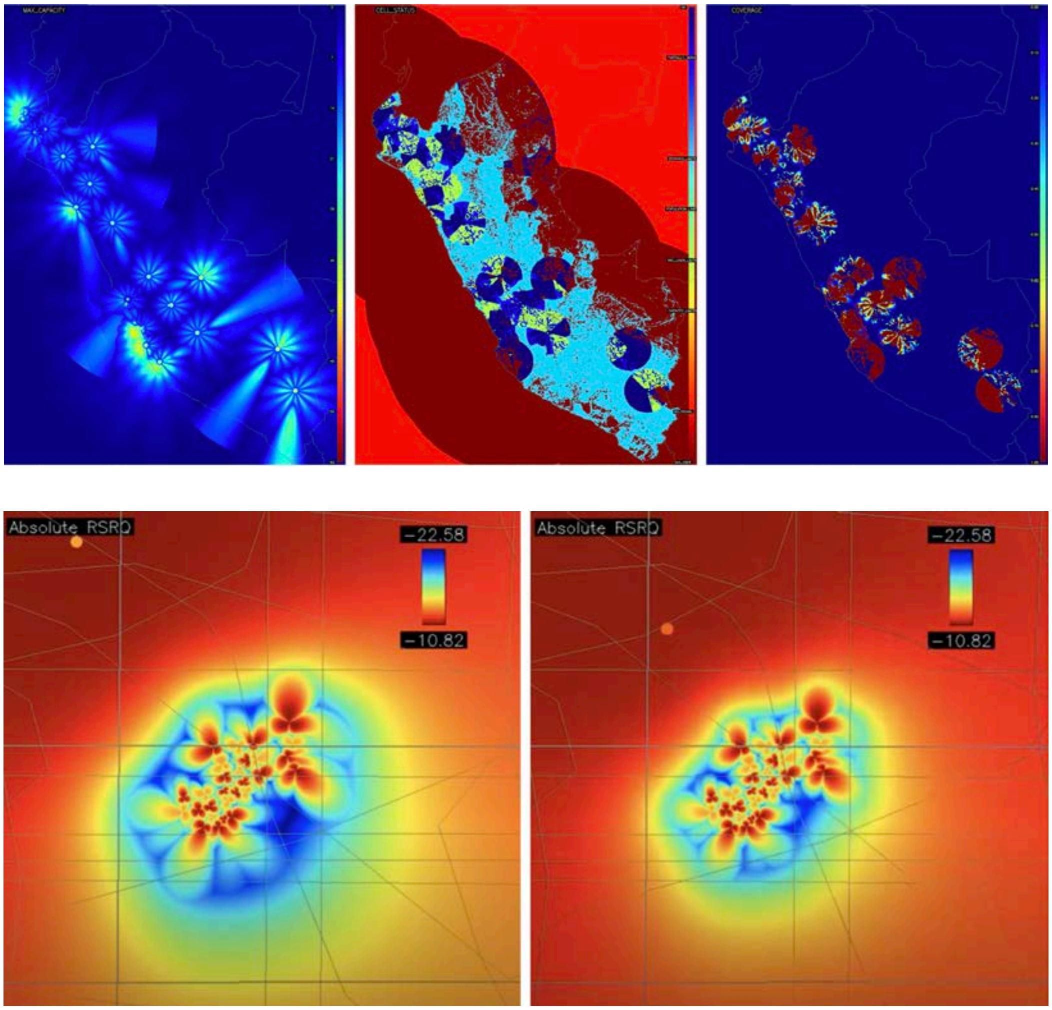


Figure 2-7 ***LTE simulator outputs.***
Top showing three key metrics for a new LTE system design over Peru. Bottom showing RSRQ (Reference Signal Received Quality) as a Loon approaches a town.

Extremely Simple Steering Model

As Loon transitioned into propelled vehicles (v1.6, with a Seahorse propeller, and a future airship design, described in "Transitioning to Stratospheric Airships" on page 351), the potential design space exploded with a dozen new design variables and a much larger selection of potential service areas. To allow the Vehicle Sizer/Optimizer to sort through tens of thousands of vehicle design candidates in hours rather than weeks, we built an extremely simple model and simulator to generate generic steering quality scores for each candidate quickly. Those scores were fed into the Sizer tool (see Figure 2-7 and "Vehicle Sizer/Optimizer Tool" on page 45), allowing it to optimize the vehicle design to optimize steering quality without requiring full VSFEI fleet sims.

Evaluating Multiple Objectives

The critical question that Systems Engineering was trying to answer with these tools is whether there is a place in the highly complex design options space where the revenue exceeds the costs and, if so, what are the specific design parameters (such as architectures, materials, sizes, and quantities) for that vehicle. What is that optimal design? And what is the definition of optimal?

The two most critical objectives for the design are maximized profit and maximized availability, but which is more important? In some markets, the availability might be essential to adoption, in which case Loon might have been willing to suffer a little bit on cost/GB. Other less picky markets might be more interested in the lowest cost service. Such two-objective problems are commonly presented on a Pareto Frontier graph, as shown in Figure 2-8. For Loon, each circle would represent a vehicle design candidate; the X-axis would show the cost, and the Y-axis, the availability. With this graph, you can see that none of the grey options are attractive; there's always at least one blue option that's better or the same on both dimensions. The curve connecting those many *best* options is called the Pareto Frontier (or Pareto front, or Pareto curve).

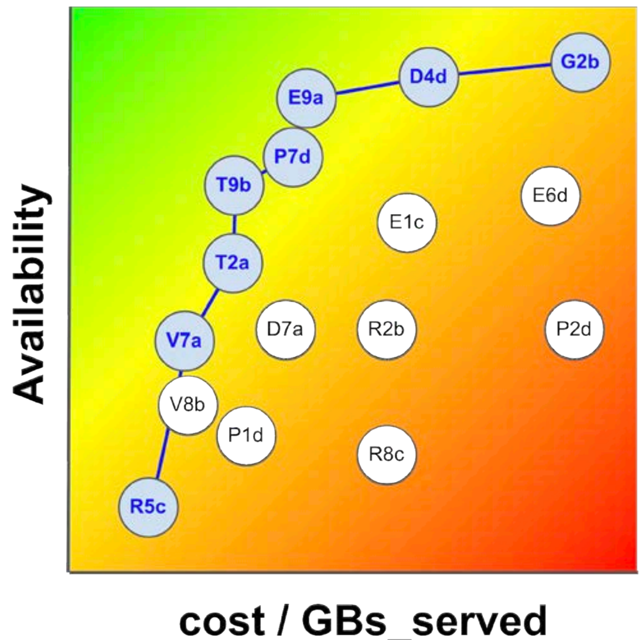


Figure 2-8 ***Loon's Competing Objective Pareto.***
Each of the vehicle design candidates along the pareto curve (frontier) might be the best choice for some market.

Loon evaluated hundreds of thousands of algorithmically created and analyzed vehicle designs with the tools described earlier. Ultimately, we ended up with hundreds of feasible (physically realizable) designs that all passed the minimum criteria for performance and cost. To compare them against each other, we laid them out on a graph like the above to determine which lie along the Pareto Frontier. From the set of design candidates that fell on that curve, Loon could select the few that were the most profitable for several different minimum availability values, such as 70%, 80%, 90%, and 95%. In this way, we can directly see that one vehicle would be optimal in the market that values availability, and another may be optimal for a market that valued price.

Unlike many other complex systems analyzed similarly, Loon's service was extremely sensitive to the differences in winds over different service areas, so this analysis needed to be performed on a per-region basis. The most profitable 80% available vehicle in one region would often be surpassed by another vehicle in a different region.

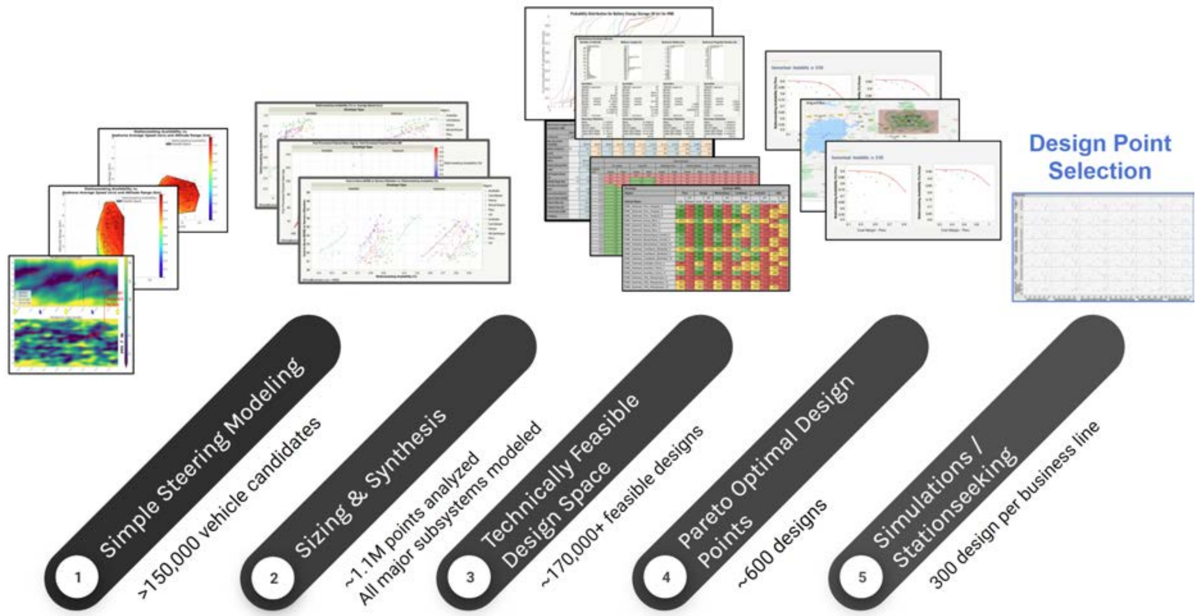


Figure 2-9 ***Loon's evolved end-to-end Systems Engineering Tool Chain and Process.*** Using these tools, new vehicle architectures and designs could be optimized to minimize cost/GB while maximizing availability and significantly increasing confidence in ultimate performance along all important business and service metrics. Shown are some candidate counts for a recent design effort.

Chapter 3

Flight Vehicle

In this section of the Loon Library we will cover the architecture and design of Loon's flight vehicles including those rolling into service in early 2021, as Loon wound-down. Future flight vehicles are described in the chapter on Loon's Trajectory and Future Impact.

Flight Vehicle Role and Challenges

The flight vehicle's role was to position the communications service payload near users to provide internet access to those users. How quickly the balloons could navigate to the users and how long they could remain there were critical factors in determining how many flight vehicles Loon would need to fulfill service requirements. Likewise, the lifetime of the balloon was the most important factor in determining how many balloons needed to be manufactured and launched in order to maintain those fleet sizes. These performance aspects of the flight vehicle had a critical impact on Loon's economics.

Achieving very long lifetimes required significant advances in the design and materials development for the balloon as well as the altitude control system, a 60,000 RPM turbo-pump that was required to operate at an extremely high efficiency for thousands of hours under huge temperature variations. As we continued to meet and increase our lifetime goals for the balloon and ACS, the reliability of the rest of the system became critical as well, including the avionics and the service payload.

Being a lighter-than-air vehicle, mass was a key focus in the design of everything on the balloon and lifted by the balloon. Being solar powered with batteries for night-time operation, extremely efficient power utilization was important since higher power use meant higher mass for the power system. The stratospheric extremes of temperature, UV exposure and very low air pressure were fixed constraints into the design process, requiring development of new technologies and innovations across the system.

Each major component of the flight system evolved over time toward higher reliability, better performance, lighter weight and improved costs. Throughout this evolution, Loon maintained a continuous focus on safety, our overall highest priority.

Loon settled on a unique superpressure balloon architecture over the past few years, nearing average lifetimes of a year with a path well beyond. The active area of focus at the time of wind-down was adding lateral propulsion to the system to improve navigation and availability and to reduce fleet sizes. Beyond this, Loon's plans extended to aerodynamically shaped airships, to introduce a step change in performance and economics. For more details, see the chapter: "Loon's Trajectory and Future Impact" on page 350.

Flight Vehicle Architecture

Using stratospheric winds, the flight vehicles were navigated to the service area, joining a fleet of other Loon flight vehicles to provide service for months. Figure 3-1 shows the fleet during service in Kenya in early 2021. In this image:

- Diamonds indicate individual flight vehicles with directional heading pointers
- Ground stations are circled in orange
- One flight vehicle in blue depicts a 25-day historical path with the predicted 5-day path in red
- Green circles are LTE sectors actively serving data to Kenyan users on the ground
- Green lines are mesh links between balloons or to ground stations

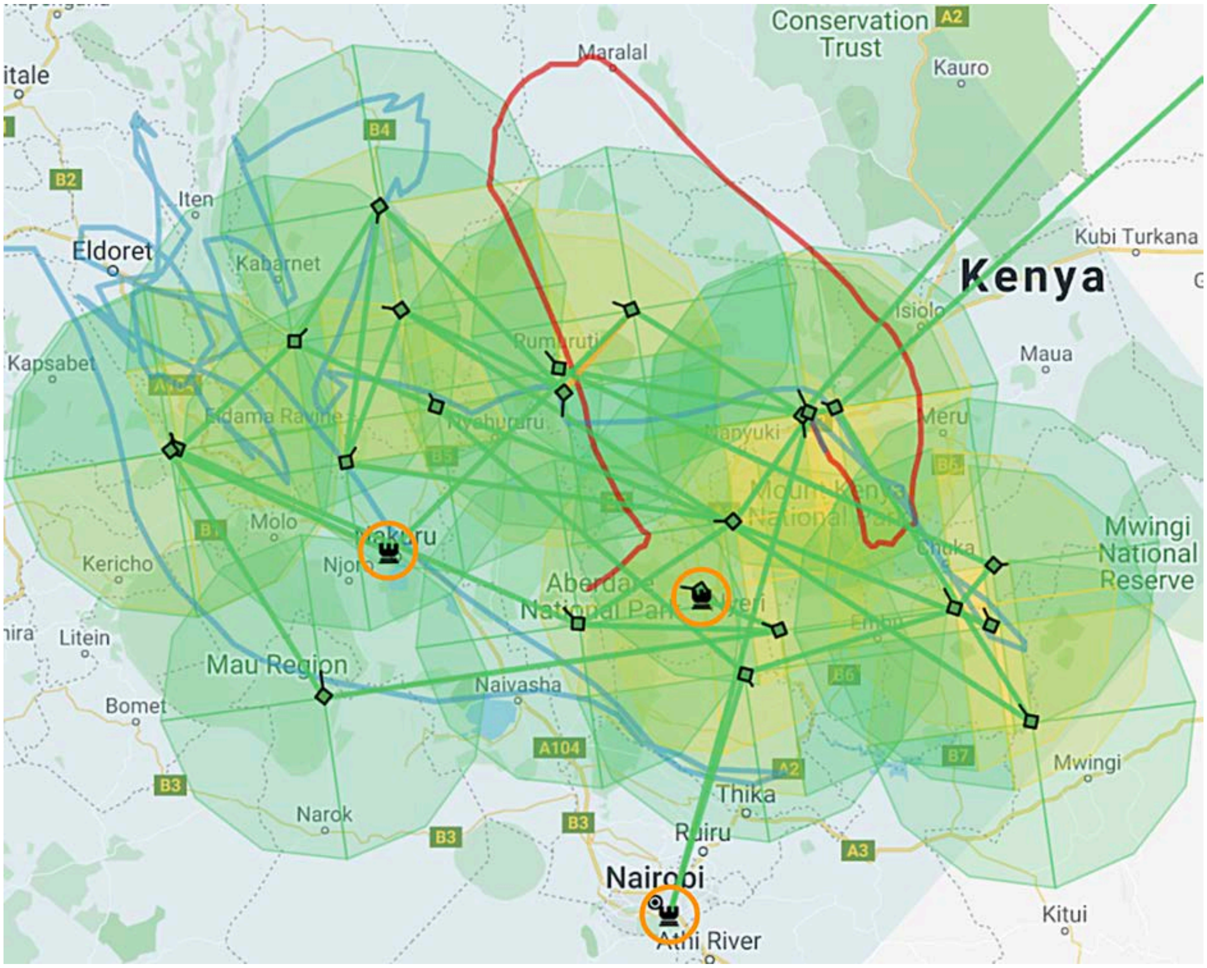


Figure 3-1 ***Fleet during service in Kenya in early 2021.***
Loon Fleet Management System's display showing balloons, ground stations, backhaul links, LTE service sectors and status and selected past (blue) and predicted (red) flight paths.

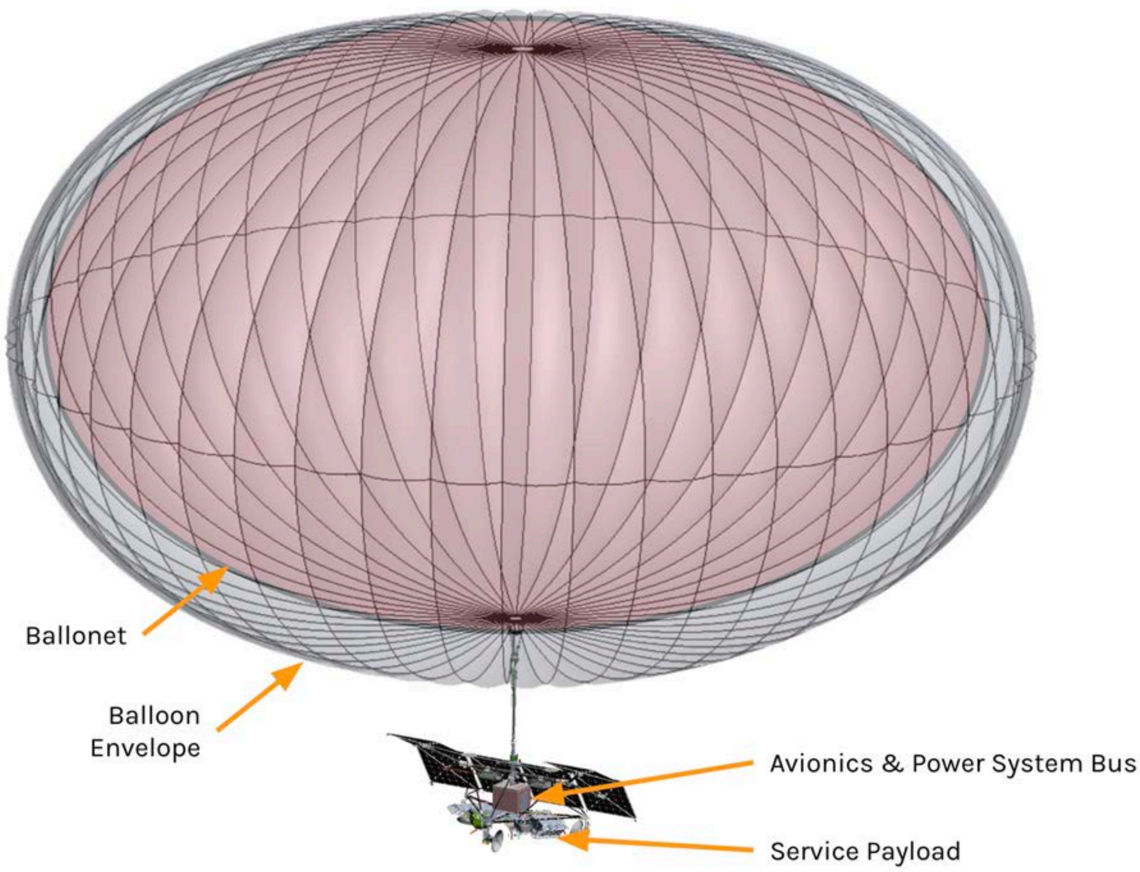


Figure 3-2 Major flight vehicle components.

Each flight vehicle consists of three major sections:

- The ***balloon***, which has an outer ***envelope*** and inner ***ballonet*** to create a helium lift gas chamber and an air ballast chamber.
- The ***bus***, which includes the avionics (all of the sensors and electronics used to launch, operate and land the system) and the power systems, including the solar panels and batteries, as well as the structure used to hold those systems and the downconnect, used to attach it to the balloon.
- The ***service payload***, which handles the mission portion of the flight vehicle. Loon's primary payload contained LTE and communications systems used to connect the balloon to the ground stations and to other balloons, and to serve the end users.

Table 3-1 lists the eight major configuration evolutions across the span of Loon, each of which delivered more in terms of capability and performance.

Table 3-1 Major balloon evolutions. (Balloon design classes were named for birds in alphabetical order)

Balloon Class	***Lark Merlin Ibis***	***Nighthawk***	***Osprey***	***Osprey Large***	***Plover***	***Quail***
Timeframe	<2015 <2015 <2016	2015-2017	2016-2017	2017-2018	2017-2021	2019-2021
Bus and Payload	Option 6	Option 6 & v1.0	v1.2	v1.3	v1.3 & v1.4	v1.6
Total flights	40+ 54+ 346	440	235	159	174 Pre-PE01 429 PE01	42 Note: Loon wound down before production launches of v1.6 Quails ramped up.
Volume Approximate Diameter	182 m^3 15 m 932 m^3 13 m 1169m^3 14 m	1273 m^3 15.2 m	1492 m^3 15.6 m	1544 m^3 16 m	1840 m^3 17 m	2760 m^3 19.7 m
Ballonet Architecture	Pancake	Balloon-in-balloon then later Yin-Yang	Yin-Yang	Yin-Yang	Reverse balloon-in-balloon	Reverse balloon-in-balloon
Altitude Range/Ceiling (usage%)	2 km @ 95%	2 km @ 95% 3 km @ 75%	2 km @ 95% 4 km @ 75%	2 km @ 95% 4 km @ 75%	4 km @ 95% 16-20 km	4 km @ 95% 16-20 km

Balloon Class	Lark Merlin Ibis	Nighthawk	Osprey	Osprey Large	Plover	Quail
Duration, average days for >20 day flights	13 41 51	53	61	85	134 (PE01) 2x @ >300 days	240 days and rising steadily
Moles Helium	4640 3050 3930	4640	5450	5720	6590	8470
Total System Mass, kg	102 67 91	98	109	112	145	192

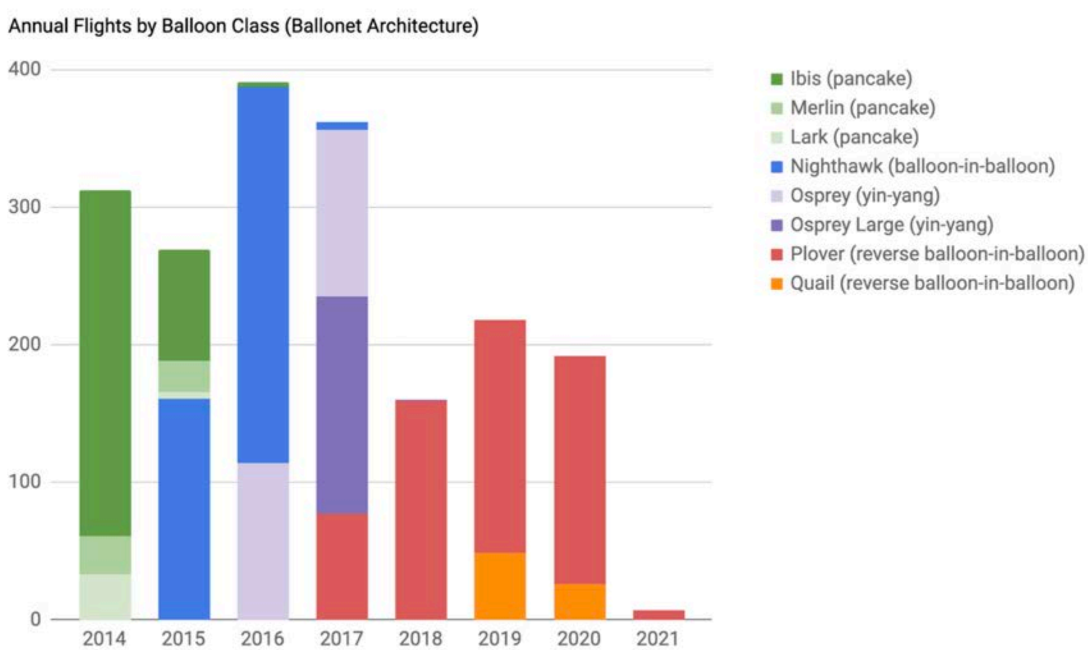


Figure 3-3 ***Annual launches per configuration over time.***
This graphic shows both R&D and production vehicles. Note that Loon stopped flights in early January 2021.

The following sections describe all major subsystems and components of the balloon and bus, with a focus on more recent iterations. More information about the service payload can be found in "Communications Systems and Service Design" on page 172.

As Loon announced the wind-down of the company, the production commercial flight vehicle was v1.4 which had been in service since 2018. By mid-2021 this was to be replaced by a new system v1.6, bringing increased navigation performance, longer life, and enhanced safety.

v1.6 maintained the same service payload but added a propeller system called Seahorse and power system updates to provide more energy for this propulsion and to increase service hours per day. Seahorse augmented the existing altitude-based navigation, enabling the flight system to stay closer to the service region for longer periods.

v1.6 also incorporated a new bus-separation descent system to deliver a number of significant safety enhancements. The bus-separation system disconnected the bus (including the service payload) from the balloon at a low altitude, to land under its own parachute. The balloon would then descend separately, also with its own parachute and including a separate set of avionics, including satellite communication, transponder, flashing beacon lights, GPS and pressure sensing.

Lastly, v1.6 incorporated a new higher-performance altitude control system enabling increased altitude range and faster ascents and descents for better navigation performance as well as redundancy to improve reliability.

Figure 3-4 ***Loon flight vehicles.***
2018 v1.4 (left) and 2021 v1.6 (right) ascending.

Balloon

The balloon itself is the largest and most visible component of the flight vehicle. It holds the helium lift gas as well as the air ballast chamber that, through varying air ballast amount and changing buoyancy, enables the system to change altitudes.

Creating a balloon that could lift the required mass and provide sufficient air ballast volume was relatively straightforward but Loon's additional requirements pushed balloon design, materials, testing and manufacturing to the extreme. Cost, performance, longevity, and quality all had enormous implications for the company so a great deal of research, development and production refinement went into the balloon.

The balloon had a number of critical design aspects:

- High strength film and tendons for supporting high superpressure for longevity (maximum helium load to maintain buoyancy) while allowing steering through changing amounts of air ballast. Changing the air ballast volume raises or lowers the internal superpressure (pressure above ambient atmospheric pressure) and therefore induces cyclical film and tendon stress.
- High quality and consistency in manufacturing to avoid leak-inducing errors over 3.5 km of sealing length per balloon. This is especially challenging given that helium molecules are so small and so are extremely hard to seal.
- Preventing degradation during long missions due to cyclical mechanical stress and environmental conditions (temperature, pressure, UV exposure) which lead to helium leakage over time, or burst in an extreme case.
- Packaging and handling to avoid introducing defects during transportation or pre-launch.

The primary structural components of the balloon include:

- Diamond shaped polyethylene gores which connect via seals along their edges to neighboring gores to form the outer envelope.
- Braided UHMWPE tendons to reinforce the gores along the seams by creating a cage-like structure to carry superpressure loads, carrying up to 1,000 kg in tensile load each.
- High-strength aluminum load rings at the top and bottom apex assemblies of the balloon anchoring the tendon ends and resisting their outward tension of 627,000 N.

Figure 3-5 ***Inflated balloon and major components.***
Also note the size, relative to the people at the bottom left corner.

Figure 3-6 ***Inflated Plover balloon inside a hangar.***

Balloon Manufacturing

Unlike most other flight vehicle components, the balloon envelope and ballonet required developing significant manufacturing advances to hit cost and lifetime requirements. This was critical if Loon was to accomplish its plan to build, launch, and fly super-pressure balloons at a rate and duration many times what had been done before. Loon partnered with Raven Aerostar for balloon manufacturing.

A great deal of R&D went into developing the optimal properties in the polyethylene film used in the balloon which had a huge impact on balloon performance. Being able to refine the film properties over time was in large part the result of Loon's extensive manufacturing and materials testing protocols. See "Envelope Gore Film Material" on page 75 for more details.

These manufacturing and materials testing protocols taught Loon many critical details about which attributes and characteristics of the film were most important for long duration flight in the stratosphere. This included aspects like strength, creep and resistance to degradation, as well as what aspects of balloon and film manufacturing processes, settings and decisions affected these material properties. These protocols included scanning the film for thickness, Instron testing (tensile strength and ultimate elongation at a breakpoint), blister testing, tendon pinch blister testing, and Gelbo-Flex testing, in which the film was repeatedly twisted and crushed at 45 cycles per minute.

In addition to this manufacturing testing there was extensive testing at full scale both in cold thermal environments and at room temperature in hangars. The ability to test at scale and take the full system to failure (burst) was invaluable in validating stress models of the balloon. In addition, it allowed for observing multiple cycles of filling the air ballast chamber which improved the design's reliability at inflating optimally. Both of these types of tests helped to monitor the effect of manufacturing tolerances when stacked up across the entire assembly.

The World's Largest Flatbed Scanner (nicknamed *Billie Jean*) was designed and built by Loon to image balloon material for manufacturing process refinement and post-mortem damage through a formal failure analysis (FA) process. It detected indications of handling or manufacturing damage or defects to the PE film such as tears, strain, and sub-millimeter holes (down to 0.2 mm). The scanner was a greater-than-room-sized (8 x 80 ft) backlit table covered with polarized glass and was long enough to scan an entire gore panel. The polarized light enabled the detection of stress and failure points and the resulting patterns were captured with a powerful, high-resolution scanning mechanism that rolled along the entire length of the table. Billie Jean used entirely customized software for scanning, viewing, making annotations, measurements, data export and reporting.

See examples of this in "Post-Flight Analysis with Billie Jean Full-Size Gore Scanning and Analysis System" on page 72 below.

Billie Jean was used for multiple purposes:

- For post-mortem failure analysis (e.g., to discover why a balloon had performed unexpectedly)
- For out-of-the-box audits to perform quality control monitoring on the manufacturing process
- For post-flight analysis of selected flights to assess the impact of balloon handling, launch/pressurization process, and degradation of the film after prolonged exposure to the stratosphere

Overall Design Evolution

Loon evolved the envelope and ballonet design several times as the team addressed both manufacturing and performance considerations, which often conflicted. For instance, a high performance design which could not be manufactured effectively was not viable. Likewise, architectures that were robust from a manufacturing standpoint but that didn't deliver sufficient performance were also not viable. See "Reverse Ballonet" on page 77 for more on ballonet design.

An early version of the balloon, known as the pancake architecture, used on the Ibis class balloon, was constructed of four circles of plastic all welded around the equator of the balloon to form the envelope and ballonet. This balloon had to be constructed as a single stack on a circular table 15 meters in diameter. As a side effect of this, the balloon was handled frequently, which increased the potential for handling-related damage that might go undetected through the Quality Control process. Additionally, scaling to larger balloons was difficult and limited, as the table diameter must also be scaled to match the balloon size. For these reasons, Loon switched to an approach using long skinny tables for sealing each pair of gores. This was used for all subsequent balloon designs.

The Yin-Yang balloon architecture separated the ballast and lift gas compartments with a hemispherical sheet of plastic film attached from pole-to-pole (top to bottom) to split the envelope into two flexible chambers. The divider sheet was assembled from multiple gores welded together. All the gores were manufactured on the same table, which reduced the total manufacturing footprint and equipment. Unfortunately, this design had challenging leak issues as well as tilt, due to the imbalance of gas densities in each side. See "Reverse Ballonet" on page 77.

Loon's 2021 commercial balloon design, the Quail, flying from 2019 to 2021, was a reverse-ballonet architecture that required the welding of 64 diamond gores together to form the ballonet and envelope while attaching load bearing tendons along the seams of the envelope. To minimize the stresses put on the film and the seams, several refinements were made to the design of the gore seam-welding tables, seam-sealing machine, and to the process for handling the partially finished envelope.

Similarly, Loon and Raven frequently evolved the process to seal and secure the envelope and ballonet material to the end plates (at the north and south poles of the balloon), secure the tendons to the load rings and to eliminate several process and material failures. Altogether, these improvements were a key enabler of the balloons' record-breaking lifetimes.

Gore Design

A major challenge in the balloon design was that at the beginning there was relatively little understanding of the stresses acting on the envelope film gores across all the different stages of life: from envelope and ballonet manufacturing, vehicle assembly, launch prep, launch, ascent and pressurization, the first few days/weeks of material equalization and throughout the balloon's service lifetime. Focusing on this gap, the balloon engineering team dramatically improved our understanding of these aspects after extensive testing, modeling and analysis.

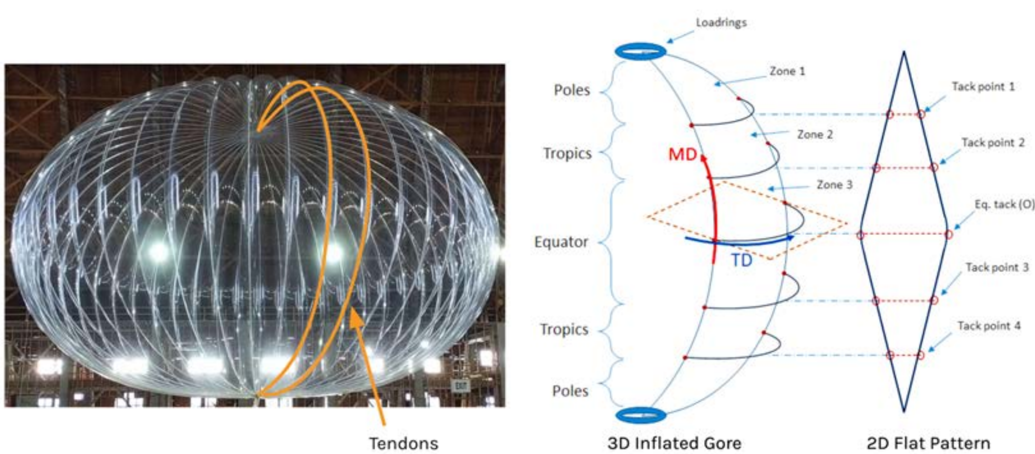


Figure 3-7 Anatomy of a gore panel.

The design optimization process for balloons started with a maximum flight vehicle mass then the optimization created outputs including balloon volume, and maximum design pressure.

The primary variables in balloon structural design for a given maximum design pressure are:

- Number of gores
- Gore geometry
- Global film *foreshortening* (difference in length between gore and tendon)
- Number of tack points
- Local film foreshortening

Tack points were small attachment features which aligned the tendons with the gore seams during filling and ascent. This greatly reduced stress concentrations throughout these processes.

These variables were estimated based on the following criteria.

- Zero stress in the meridional direction (MD, or north/south) in the polar regions and negative MD strain
- Reduce stress in transverse direction (TD, or east/west) and vertical (MD) directions
- Reduce pinch/wrinkle size in critical locations

The stress/strain state in the balloon gores varied by location as seen in the finite element analysis (FEA) plots in Figure 3-8.

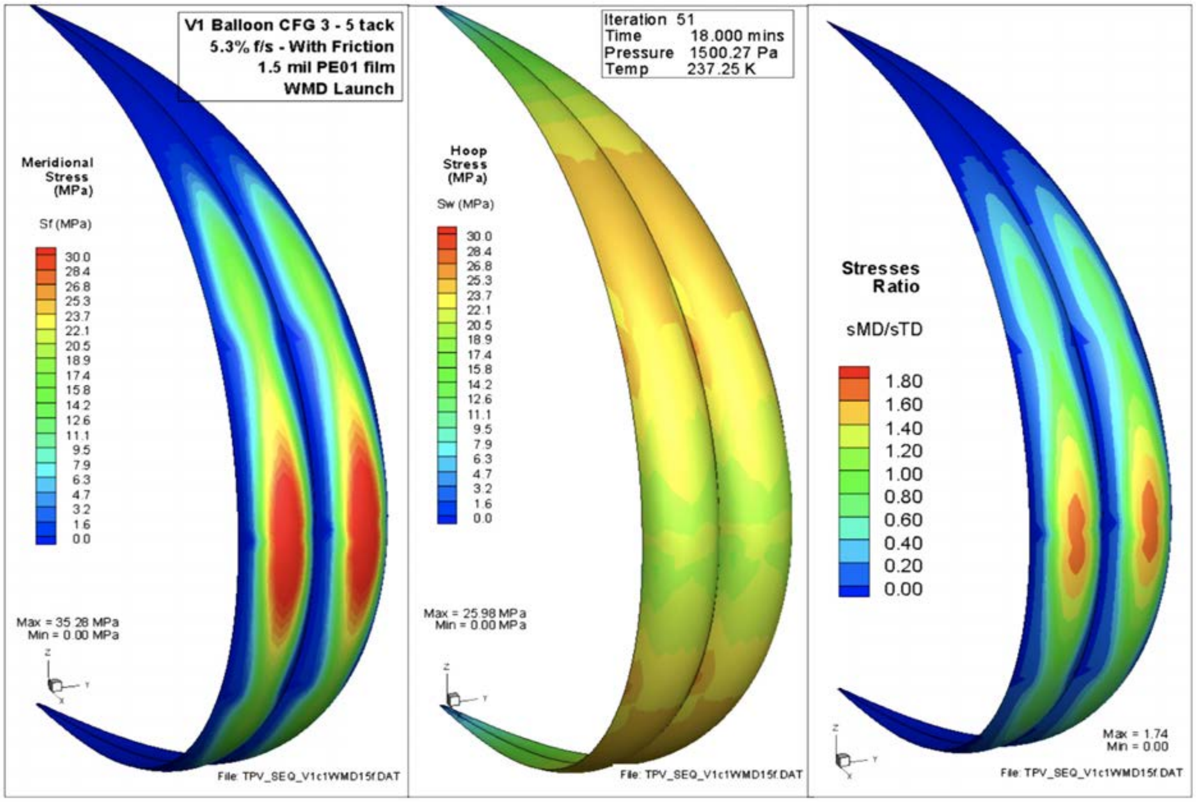


Figure 3-8 ***Gore design with FEA.***
Stress and stress ratio in FEA contours at 1500 Pa.

Figure 3-9 ***Envelope-only balloon in-flight.***
Gore wrinkles shown in the fully inflated state.

The balloon volume was established by the shape formed by the tendons in a fully inflated state also called the tendon cage. This established the tendon length. The volume of the gores bulging under pressure accounted for only ~5% of the total volume.

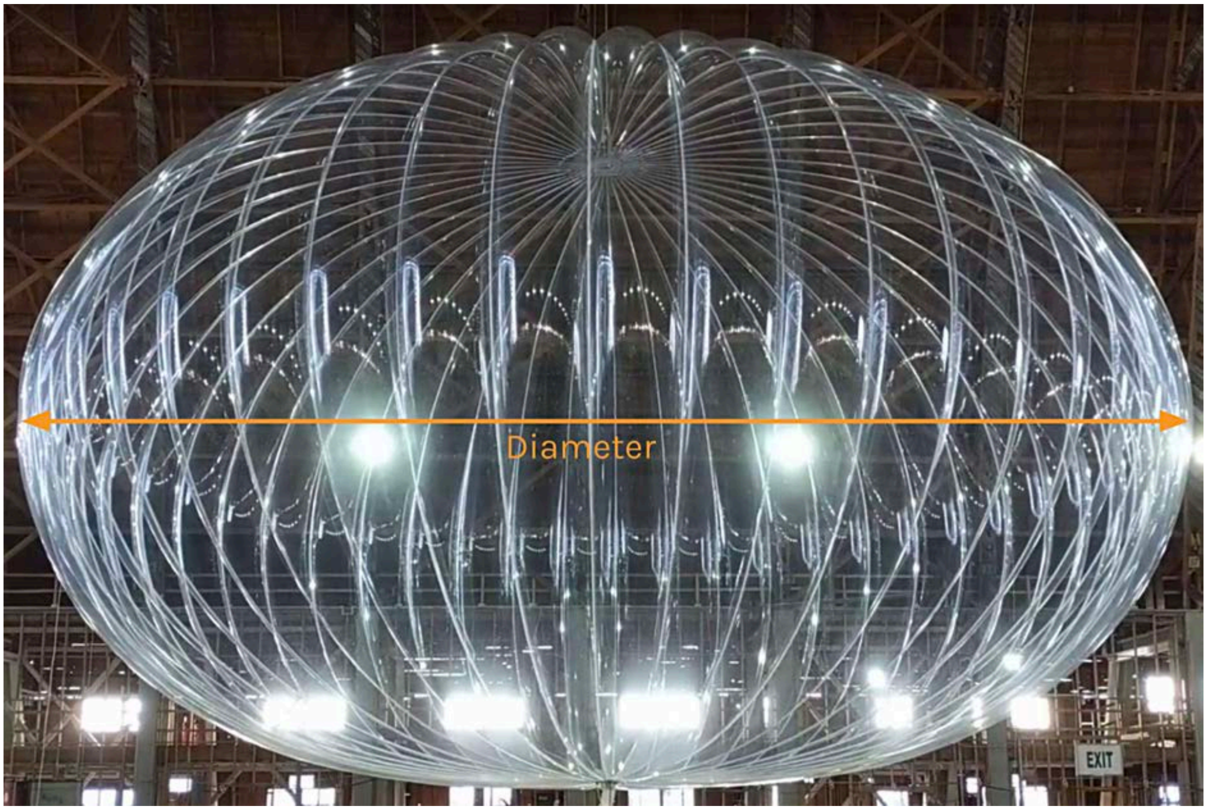


Figure 3-10 ***Envelope-only balloon inflated in a hangar.***
At sea level, indicating characteristic diameter for the pumpkin shape.

The gore width is limited by the availability of film sizes that can be produced but not necessarily limited in length. To accommodate the limitation of available width, the number of tendons can be increased such that the circumference/width criteria is fulfilled by the number of gores.

The balloon design pressure requirement therefore yielded a balance between gore width and the number of tendons and their break strength.

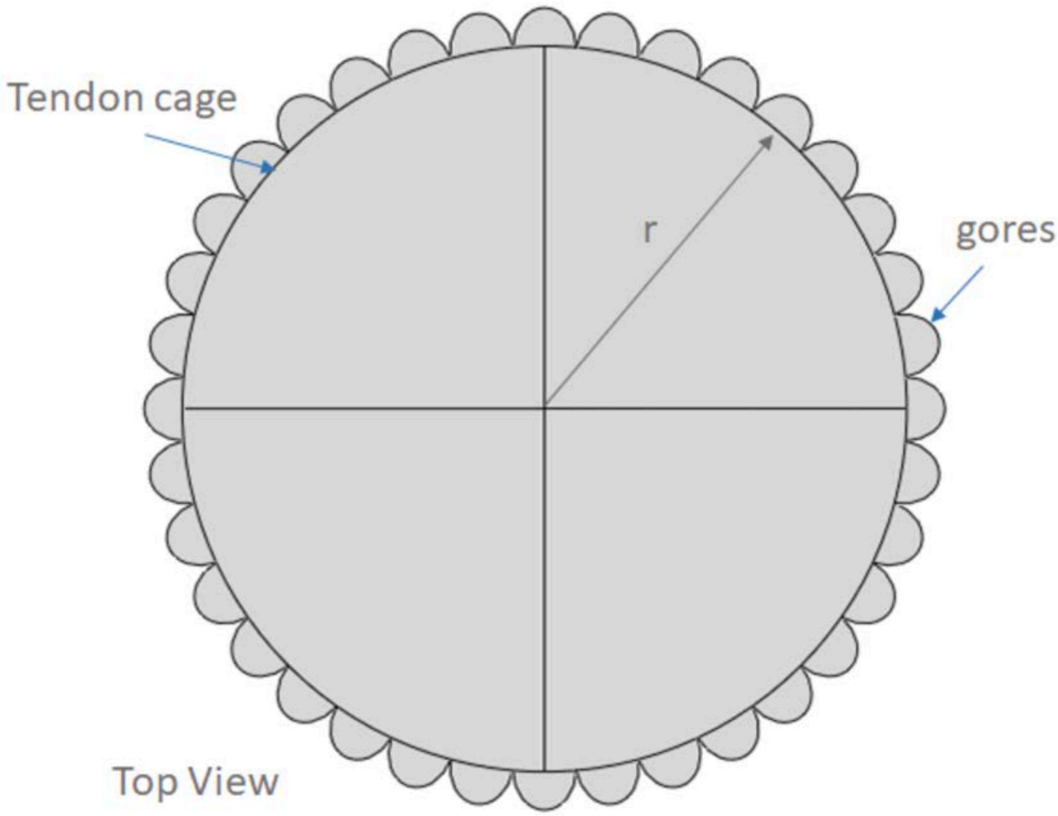


Figure 3-11 ***Top view of pumpkin shape.***
Tendon cage and gore bulges are shown.

The force carried by each tendon is calculated using $F=P\times \pi \times r^2/n\_tendons$ where *P* is the superpressure in the balloon, *r* is the balloon radius and *n_tendons* is the number of tendons.

The tendons were made of a braided UHMWPE Dyneema (DM20) while the film used was a formulation of polyethylene film chosen for the best properties. See "Envelope Gore Film Material" on page 75 for more on the specifics of the film development and selection.

Figure 3-12 Braided DM20 tendon attached to the apex assembly load ring.

The braided tendons changed length quite significantly due to tensile loading as the balloon superpressure changed. Temperature swings also changed the length relative to the gores given differences in Coefficient of Thermal Expansion (CTE).

The gores changed length much less as they were not as affected as the large fluctuations in tension in the tendons. Managing the length difference between the variable tendon length and the less variable gore length was a factor called foreshortening. This was effectively choosing the right length difference range between gore and tendon over the operating pressure and temperature range and had large effects on the envelope behavior and stresses. Since the difference in length could result in excess stress or excess film material to be managed and potentially damaged, it was a key factor in the balloon design.

Throughout design exploration and testing, it was established that a foreshortening value of ~3% was not adequate enough and ~7% was too much and was prone to pinch formation and risk of instability.

Foreshortening was calculated based on tendon length and gore centerline (or edge) length.

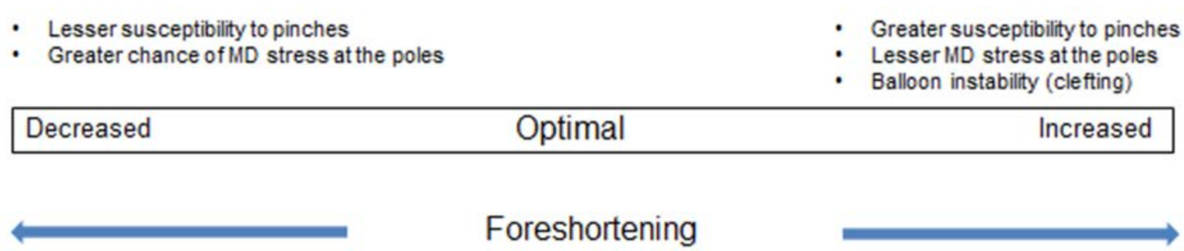


Figure 3-13 Tendon foreshortening optimization.

The gore pattern could result in different stress states along the length of the gore based on the types of patterns used for sizing either constant angle, or constant radius. For gore patterns, see Figure 3-14.

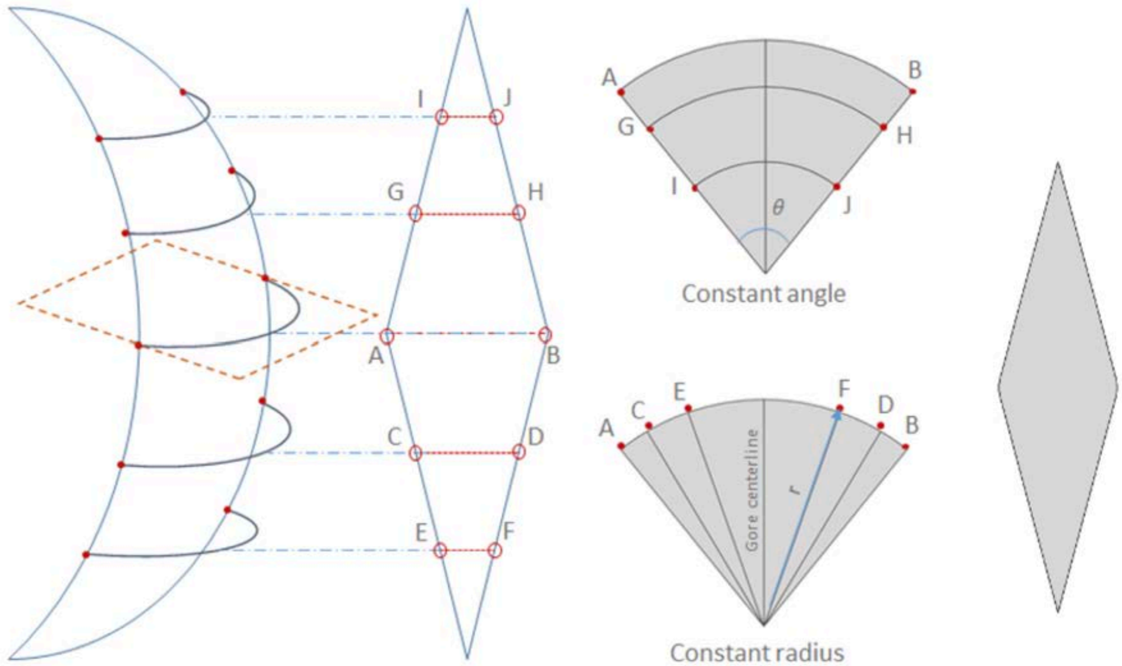


Figure 3-14 Gore pattern.

Diamond Pattern

Although a gore would appear to be a curved-sided shape, Loon (and others) have used diamond shapes instead, both for manufacturability and to improve the distribution of stress across the film.

The major dimensions of a diamond pattern gore were based on structural and manufacturability requirements such as:

- Gore length: centerline length based on global foreshortening
- Gore angle: Ø is 360/n_tendons
- No. of tendons could be a function of available gore width
- Gore cut radius: R is based on manufacturability and is 50 ft. for Plover design

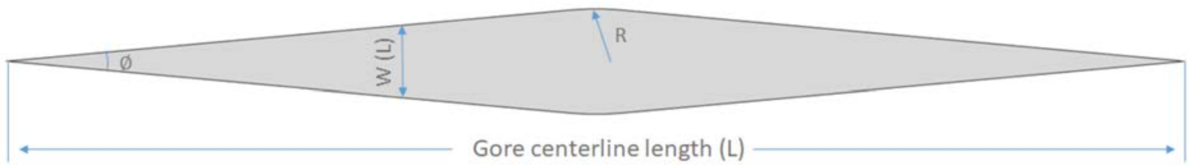


Figure 3-15 Gore diamond shape.

Optimized Stress Profile Sealing

Pattern selection also considered non-structural constraints such as ease of manufacturability. With the existing diamond table, the latest generation sealing system allowed for variable geometry which allowed for exploring different designs to optimize stresses and strains.

Post-Flight Analysis with Billie Jean Full-Size Gore Scanning and Analysis System

A central tool in the engineering of the gore panels and tack point locations was post-flight analysis of gores scanned via the full-gore-sized Billie Jean light table. Full gores from flown balloons could be laid out on the table and scanned for detailed imagery to classify location and temperature conditions of failure points. This tool was a major contributor to balloon engineering progress.

Figure 3-16 ***Billie Jean full size scanning light table.***

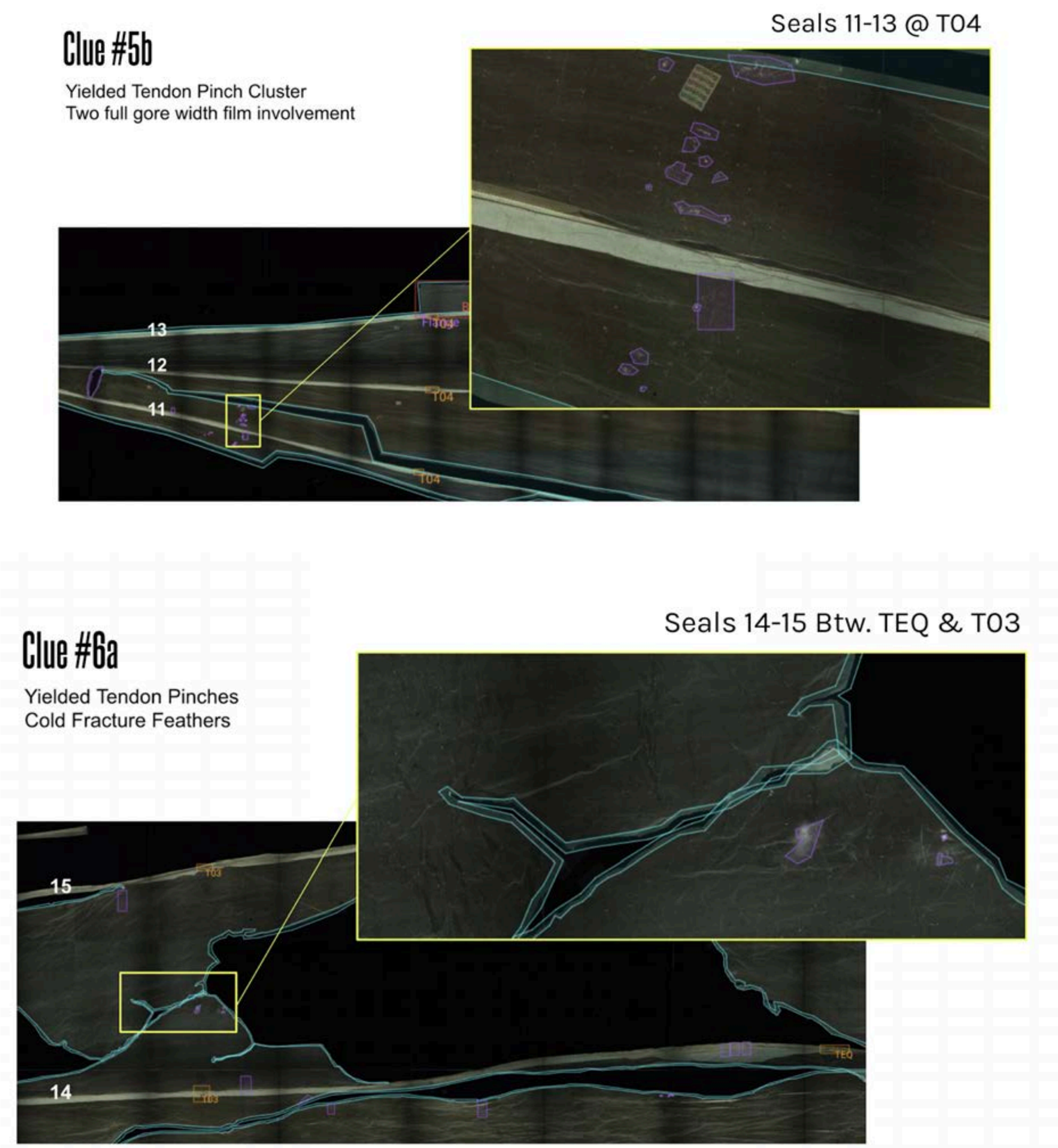


Figure 3-17 ***Example Billie Jean imagery and analysis scan.***
Rapid depressurization progression on a test flight are evaluated.

Envelope Gore Film Material

The envelope film material itself turned out to be a critical area for achieving long mission durations. Aside from its importance in basic strength it needs to resist degradation when exposed to the cyclical mechanical, thermal and UV radiation environment of the stratosphere for up to a year. Thus, film material selection and manufacturing refinement was an area of tremendous R&D focus.

Over the years many versions of polyethylene were tested, both in labs and in real flights, trying to maximize different properties. From 2016 to 2018, the film choice stabilized on a version with the best overall properties of those Loon had tested. Then in 2018, after extensive investigation and co-development with a film partner, Loon transitioned to a custom 1.5 mil blown polyethylene which Loon named the PE01 material. This was a cross-linked polyethylene with increased orientation compared to the previous generation film. The two films had different moduli and CTE with PE01 bringing higher strength and better resistance to degradation.

The higher strength reduced the operating stress for the same load which resulted in improved resistance to creep which was one aspect of long term degradation. In addition to this, the film had increased optical clarity which lowered operating temperatures and slowed chemistry-related material degradation. An added benefit; it made for some gorgeous crystal clear balloons.

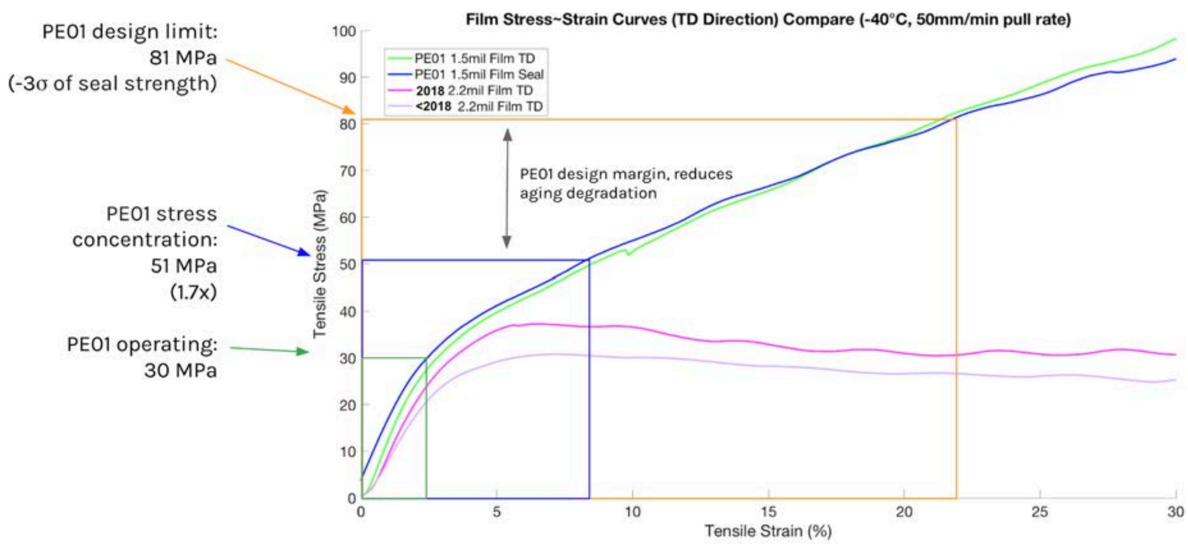


Figure 3-18 ***Stress-strain curves for the pre-2018 film, the 2018 film and PE01.***

These material properties added significantly to the balloon operating lifetimes achievable. Operating lifetime limits were established for all films to define a high confidence operating lifetime during which Loon was highly confident degradation was within acceptable tolerances. Over time, this limit would increase as more balloons provided more data points to test and inspect to characterize degradation.

At the time of the transition to PE01, the previous film had established a high confidence operating lifetime of 110 days. At the time Loon wound down PE01 had already increased the high confidence operating lifetime to 175 days and this was still increasing. Data indicated it would go much higher and Loon already had a number of balloons which exceeded 320 days lifetime by the time of the wind down.

This improvement in properties came about thanks to a very close collaboration with the film manufacturer including process development. It took many trials to explore the parameters needed to achieve the potential of the strength increase. This co-development turned out to be essential.

Apex and Bottom Assemblies

The balloon has two assemblies at the top and bottom point where all the tendons and gores converge. These two assemblies fulfill a number of roles for the balloon and allow for mounting a number of different systems:

The Apex assembly includes:

- High strength aluminum load rings to support the full outward tension of the tendons by securing the end of each tendon to a post
- Mounting points for attachment to the Lift and Fill Rig (LFR) used for holding the balloon up while filling
- Feature plate including donut seal which seals all the gores of the envelope and ballonet together to a rigid plastic plate. The feature plate incorporates the following:
  - » Windows for Flight Termination System – See "Flight Termination and Descent Systems" on page 122 for details
  - » The FTS cutters themselves
  - » The avionics board responsible for controlling FTS triggering
  - » The balloon fill port

  - » A lightning cage to prevent electrical activity from affecting the FTS
  - » A satellite communication system with backup battery system to mitigate command and control outages

The bottom assembly includes:

- High-strength aluminum load rings (as on Apex)
- Altitude Control System (ACS)
- Three-legged campstool structure to connect the downconnection to the bottom assembly
- Launch handles which lock the filled balloon to the launch cart. These are the last attachment point released when launching a flight vehicle

Reverse Ballonet

Loon experimented with a number of approaches for separating the lift gas chamber from the air ballast chamber. The objective was to improve mass-efficiency, altitude range (how much air could be pumped into the ballast chamber as a percentage of the total balloon volume), manufacturability, quality, cost and preventing leaks.

Approaches fell into two architectures: using a diaphragm to split the envelope into two chambers and using a balloon-in-balloon approach. After exploring iterations of both, Loon developed a version of the balloon-in-balloon architecture, which, while not as mass-efficient as the previous Yin-Yang architecture, met all of the objectives, but now with a dramatic improvement in lifetime due to mitigating helium leaks. All previous architectures had an inherent pressure-differential-driven helium leak mechanism that had limited lifetime due to the pressurized helium chamber being exposed to the low pressure of the stratosphere.

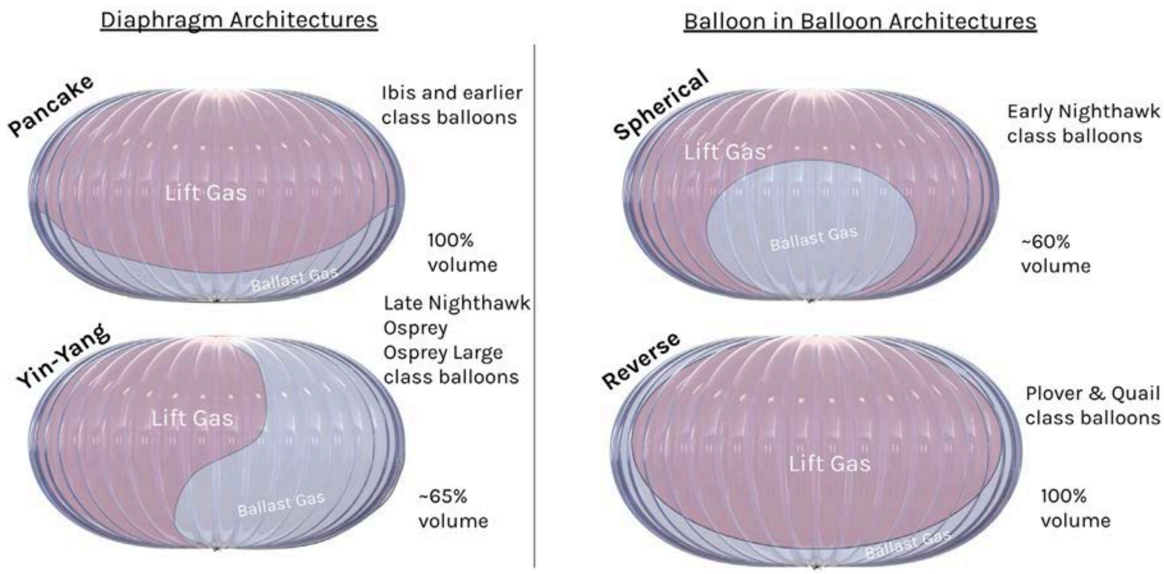


Figure 3-19 Ballonet Configurations.

This pressure-driven helium leak issue was solved by the transition to a reverse ballonet configuration. This was a deviation from the dominant superpressure balloon configuration used over decades, where the lift gas was in the outer chamber and air ballast was in the internal chamber. Loon reversed the arrangement of the gasses in the new configuration. The key benefit of this was that the barrier between helium and air was inside the pressurized shell and so did not have a pressure differential driving leaks through the film or seals.

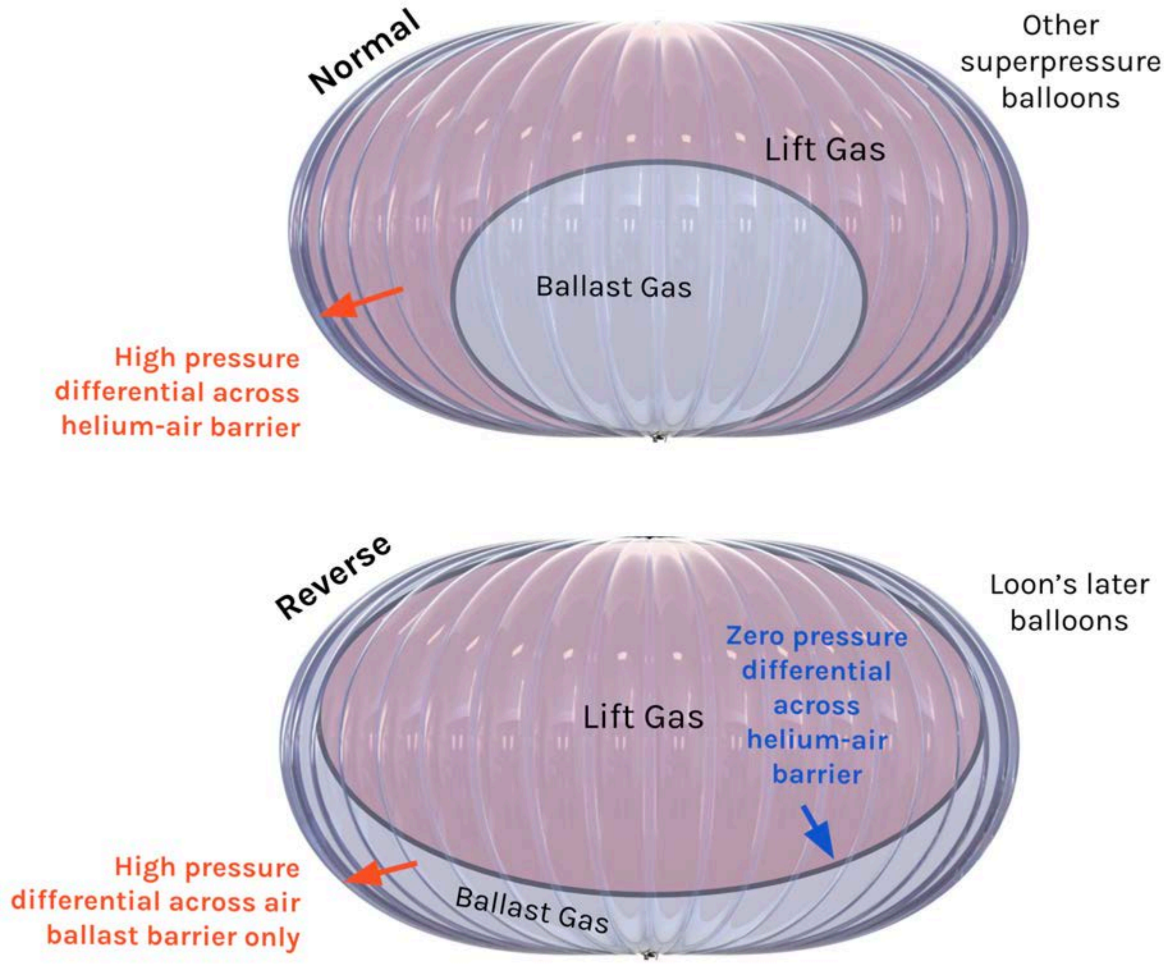


Figure 3-20 Pressure differential locations in normal vs. reverse ballonet.

Advantages of the reverse ballonet architecture:

- Dramatic increase in lifetime due to leak reduction as helium-air barrier has no pressure differential
- Ability to use Helium Launch Assist (HeLA) process to reduce stresses between film layers

Disadvantages:

- Difficulty in filling the ballonet which is attached at the top since it must be filled from above at the apex assembly

Balloon Design Statistical Performance Comparison

Loon applied statistical methods to evaluate balloon designs for both unforecasted failures (including all types such as helium leaks, ballonet leaks, and high degradation rates) as well as lifetime. The statistical comparison plot below is from April 2019 and shows how the effects of the reverse ballonet and the material change to PE01 affected both. Of particular note is the influence of PE01 on both failures and lifetime even within the Plover reverse ballonet design.

By the time of this plot in April 2019 only 30% of all PE01 Plovers had been launched and characteristic lifetime continued to improve with the increased flight data.

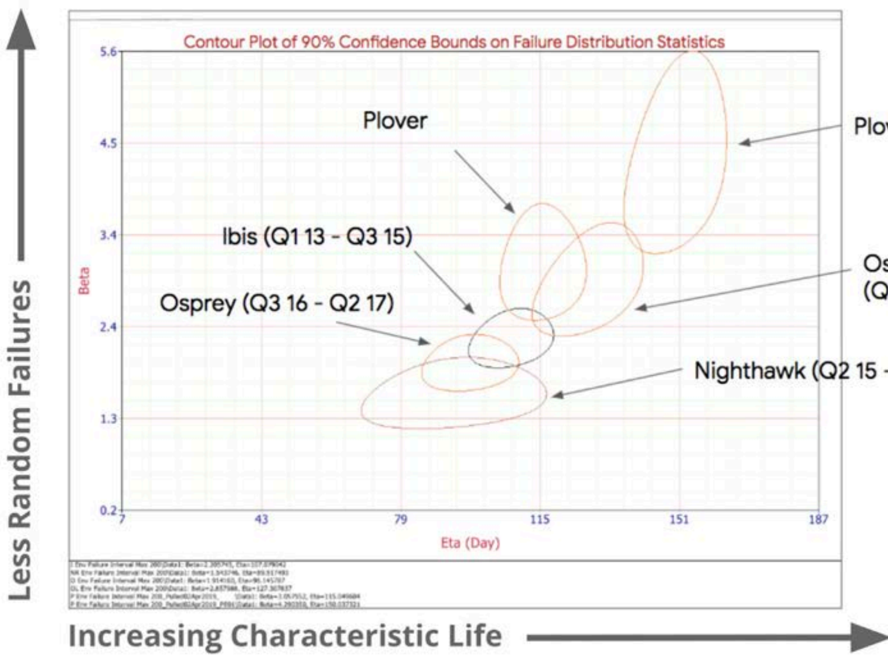


Figure 3-21 ***Statistical comparison conducted in 2019 of balloon designs over time.***

Helium Launch Assist (HeLA) Process

Loon developed a useful launch process that the reverse ballonet configuration made possible. The Helium Launch Assist (HeLA) process injects some helium into the air ballast chamber through the bottom assembly prior to launch, both to improve ascent and to help reduce potentially damaging stress to the film when the main lift gas chamber is filled. As the balloon approaches the float altitude, the helium is pumped back out of the air ballast chamber.

The potential damage to the film can occur due to the way the balloons are packed after manufacturing. Packing the envelopes for shipment with no air inside either of the chambers ensures that only helium, and no air, is present after the fill is complete. This creates tight coupling of the film sheets together which can lead to stresses between those sheets while the balloon is filling and pulling upward on the launch system. In the extremes this can lead to pinching and 'witch hatting' where the film is pulled and stretched into a tight pinch; these can then become leaks between the lift gas and air ballast chambers.

Loon modified a previous similar procedure to use helium instead of air. This meant the gas would propagate upwards during fill to separate the films at the top of the balloon where most of the tension was generated. Once at float this helium could be vented by the ACS to allow for a full chamber of air ballast.

This not only helped reduce damage to the film, but it also increased free-lift buoyancy for a more rapid ascent rate or a normal ascent rate with a heavier vehicle, for example, carrying extra powder ballast. (See "Powder Ballast for In-Flight Reducing of Flight Vehicle Total Mass" on page 106.) Rapid ascents provide several advantages: reducing downwind drift, decreasing the risk of a stalled ascent and minimizing the impact of controlled airspaces.

Figure 3-22 Balloon filled prior to launch.

Summary of Key Balloon Learnings and Innovations:

- Manufacturing, R&D, and testing play critical roles in developing a reliable balloon and should be viewed as an on-going effort.
- Refining film properties over time can deliver significant performance improvements. Collaborative codevelopment with the film producer is key.
- Reducing leaks through reverse ballonet design is a key enabler of very long balloon lifetimes.

Altitude Control System

At the bottom of the balloon is the Altitude Control System (ACS), which pumps air into the air ballast chamber or opens a valve to allow it to vent out. This enables the balloon to ascend or descend between specific layers of the stratosphere, following the different wind directions needed to steer the balloon to desired locations.

Air ballast systems are extremely efficient and work well for moving through different altitude ranges in the stratosphere. Electric turbomachines are a perfect fit for the compression of air into the ballast chamber, but they must be sized appropriately for the vehicle, and with appropriate operating design margin. While the reliability requirements should be planned per a considered reliability budget, ideally this critical system has at least partial parallel redundancy, as developed for the v1.6 Loon flight vehicle in 2021.

Loon's ACS consists of two main elements:

- The compressor which pushes air into the ballonet against the superpressure of the envelope.
- The valve which seals air ballast inside against this internal pressure and also allows for venting air ballast out of the envelope to ascend the flight vehicle.

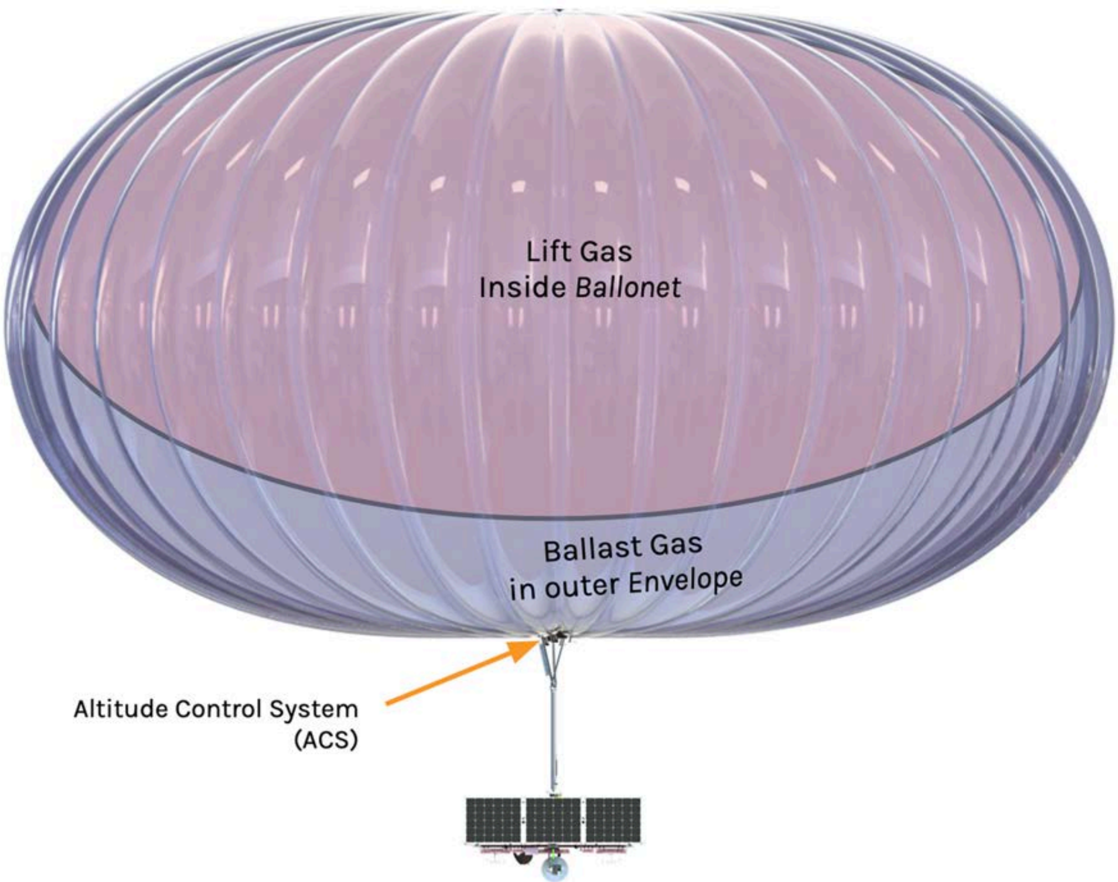


Figure 3-23 ACS location on the flight vehicle (integrated into the bottom assembly).

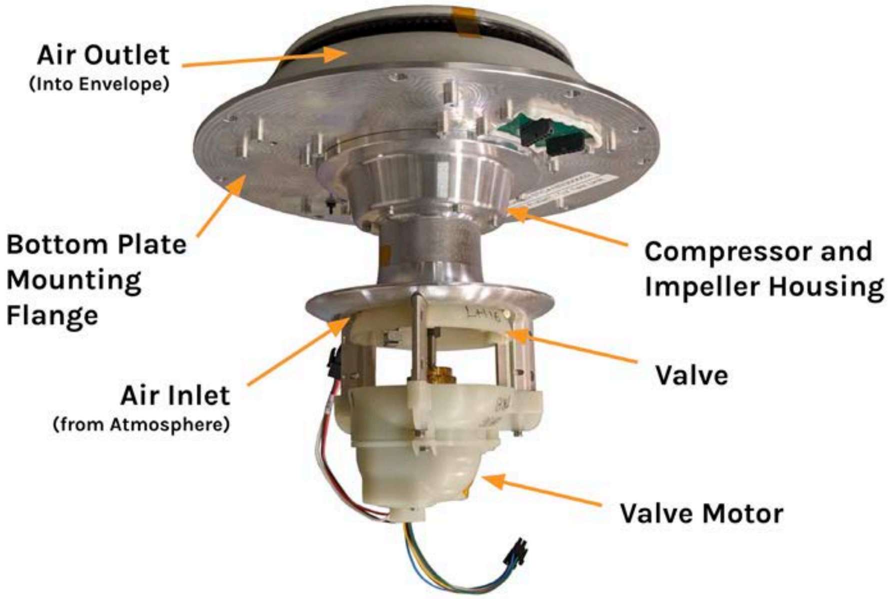


Figure 3-24 ACS compressor-valve components.

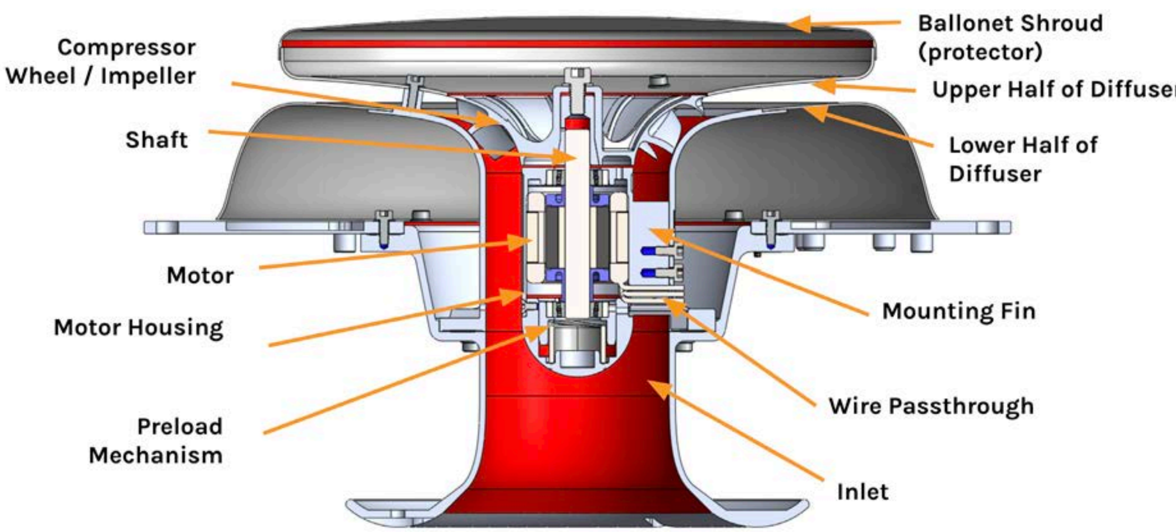


Figure 3-25 Major components of the ACS compressor.

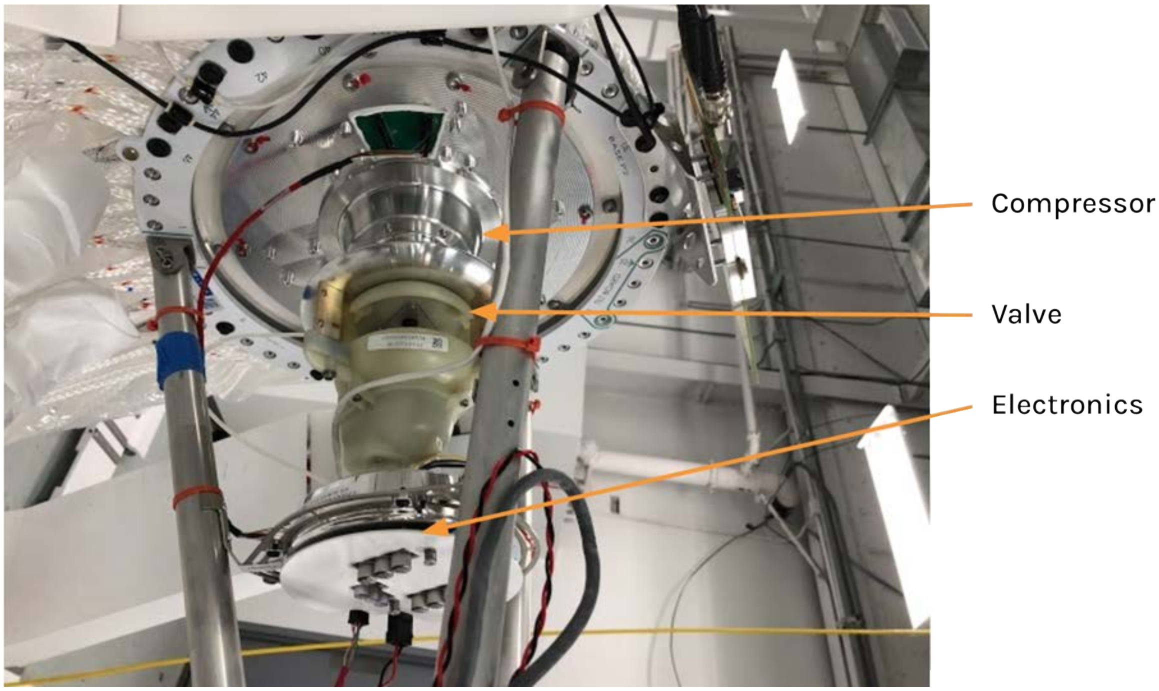


Figure 3-26 Major ACS components mounted for flight on the bottom assembly and campstool.

Loon explored different ACS approaches over time with a focus on multiple areas of improvement throughout stages of the project. Early systems were driven by providing simply the functionality to explore the feasibility of using winds to steer the balloon. When this was proven to be viable, the focus shifted to both reliability and power efficiency for the majority of the project, with generational leaps with the Franz 3.0 and Thor designs. Once Thor was developed and refined, it proved to be an extremely efficient and reliable compressor (with some units exceeding 300 days of life). After Thor, the focus shifted to increasing vehicle agility (the speed at which it ascended or descended) as it scaled in size, as well as adding partial parallel reliability by integrating two Thors into a single ACS.

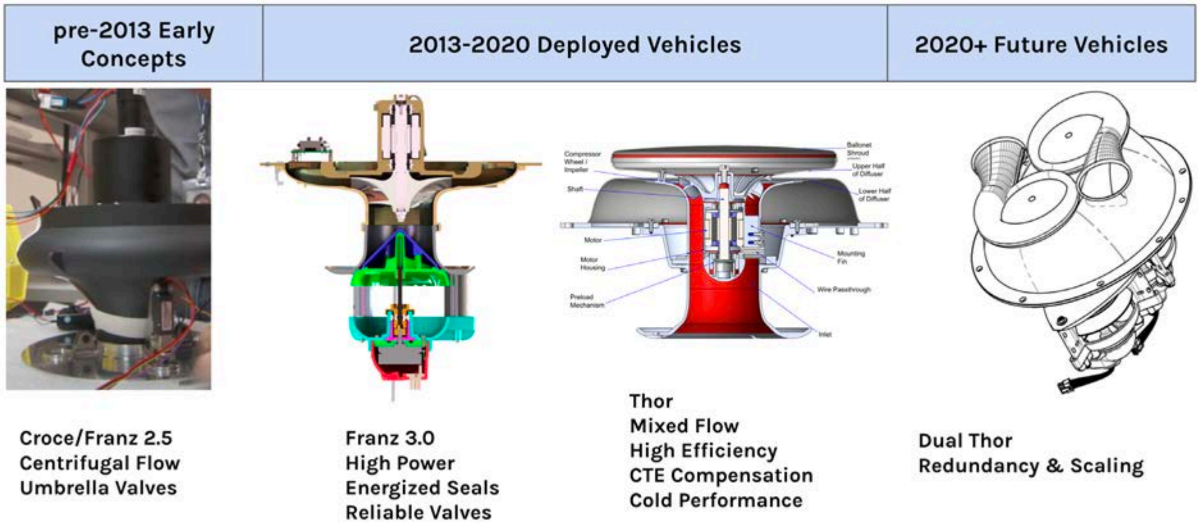


Figure 3-27   ACS design evolution high points.

ACS Development Evolution

As the entire flight vehicle continued to evolve, Loon's ACS design and operational challenges included:

- More effective steering algorithms, especially those based on reinforcement learning (RL), led to very high cycle counts for a typical vehicle lifetime (eventually, > 6,000 compressor actuations, > 12,000 valve actuations, > 1,000 hour expected compressor runtime). Though further optimization of these algorithms would reduce the cycle counts, it was clear that higher cycle counts translated to better navigation.
    - Especially with regards to cycle count (which grew to 10x initial design life expectations), this exacerbated initially unknown failure modes for the compressor (rotordynamic system oscillations and preload mechanism failures) and sealing mechanism failures in the valve (debris generation from both wear and compressor-coupled vibration).

- Multi-objective design optimizations for efficiency, mass-flow rate, and reliability drove sharply conflicting requirements. The custom nature of the solutions required highly analytical design approaches for the bearing size and arrangement, turbomachine design, and motor design. After success developing a design with the desired efficiency, there was higher susceptibility to small variations in components, occasionally causing early-life failures.
- Realistic reliability testing (emulating the highly variable environment of stratospheric low pressure and thermal conditions) was difficult to implement on the ground, and large sample size testing was not yet implemented. As discussed in the reliability section, this masked defects of unknown failure modes in production without HASS (Highly Accelerated Stress Screens) or large sample ORT (On-going Reliability Testing).

Critical ACS Milestones and Lessons Learned

This section details critical ACS milestones and lessons that the Loon team learned during the Loon project lifetime.

Passive Thermally-Compensated Bearing Preload Systems

Loon's electric turbomachine ACS was power dense and operated under dynamic heating-cooling cycles (worst case for a cold start had a range from -100°C at start to +50°C once operating). It was difficult to match the different material's coefficients of thermal expansion without sacrificing one of the major requirements noted above. This proved especially problematic for fixed bearing preload systems with limited travel, where these mismatches led to both overloading and underloading in the bearing system. These conditions would severely affect the rotordynamics or bearing fatigue life. For purposes of longevity in high speed rotating systems, ideally light, but consistent preload is required if efficiency and system stability is to be maintained. An intermediate version of the preload system is shown below depicting the moving end of the specific machine's angular contact bearing pair. In this machine, the preload bearing element translates in the direction of dimensional change from thermal changes (nonlinear growth or contraction). Concentric O-rings or other energizing elements could be used to provide mechanical stiffness if required by the rotordynamics model. Flexure-based bearing element mounting was considered an ultimate solution for both the stiffness and translation required, but fitting into the space available, specific implementation, timeline considerations, and manufacturability issues precluded integration into production before Loon wound down.

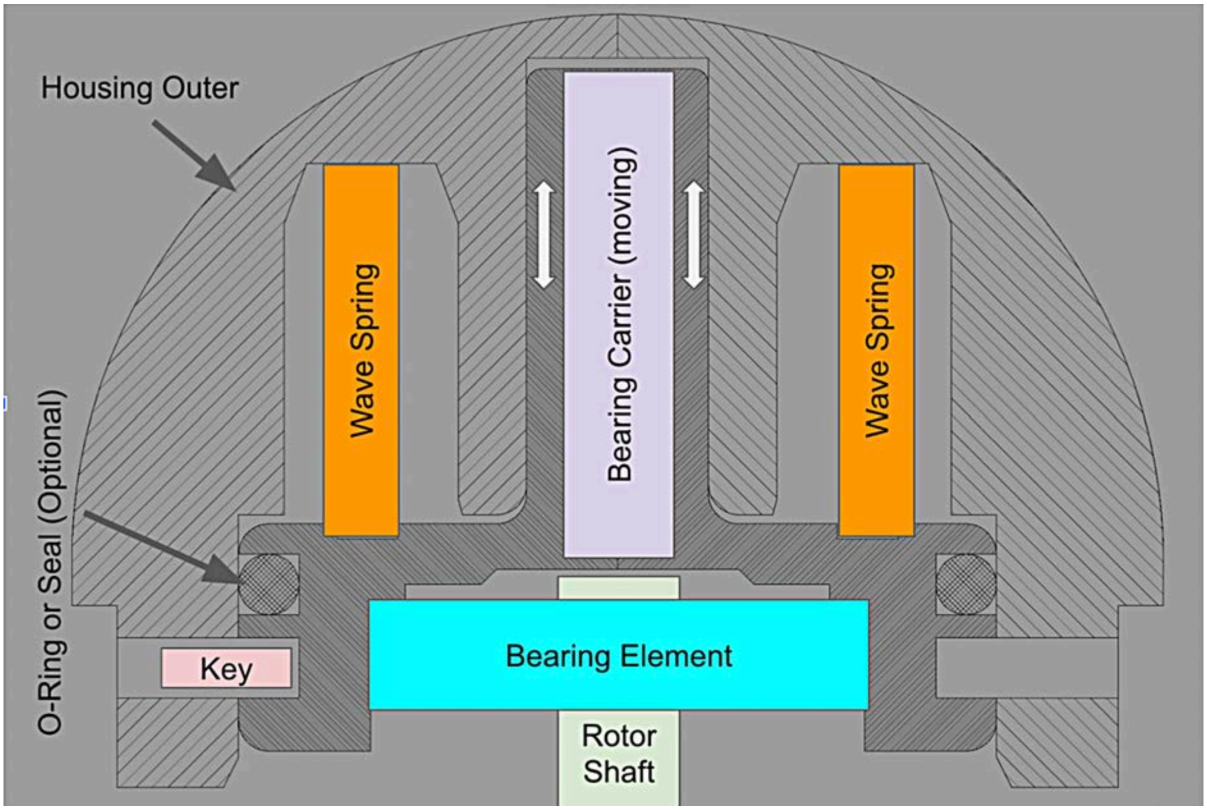


Figure 3-28 ***Example moving preload system.***
Moving side of bearing pair is shown.

End-of-Line Spectral Fingerprint of Vibration at Time Zero

Typical for most high speed systems (Thor ran at speeds up to 60,000 rpm), per-part accelerometer vibration analysis was performed on compressor systems once built, with known good device signatures cataloged against results from reliability testing. Manufacturing defects in the system could potentially be diagnosed against these spectral signatures. Since systems cannot be inspected once flying, the use of End of Line tests for characterizing vibration was extremely helpful in correlating in-flight issues and providing eventual screening or sorting methods. Note that while this tool is useful as a screening method, the root cause for the defects noted by the process may have multiple sources, such as misalignment, preload failures, or defective bearings. It's also important to note that not all failure modes were captured in this benchmarking process, as evidenced by some of Loon's early mortality issues with some compressors even later in the program (for this large sample ORT and HASS could provide coverage. See "Loon Reliability Concept Recommendations" on page 169.

Optimized Aero-Rotors and Electric Motors

Power and mass are at a premium in Loon's flight vehicle; maximizing efficiency of the ACS is a critical system requirement. Creating a high-efficiency (>70% isentropic efficiency) turbopump requires sufficiently high quality FEA grids and accurate boundary conditions to design a machine that meets performance requirements. Failure to do so (as was done in early designs) means leaving off 30-40% of potential specific mass flow (which was measured in watt-hours per kilogram of air moved). Partnering with an expert in these fields is highly recommended. Additionally, consideration of boundary separation and flow is essential in these analyses to model mass flow over the pressure ratios required for operation to obtain an accurate model.

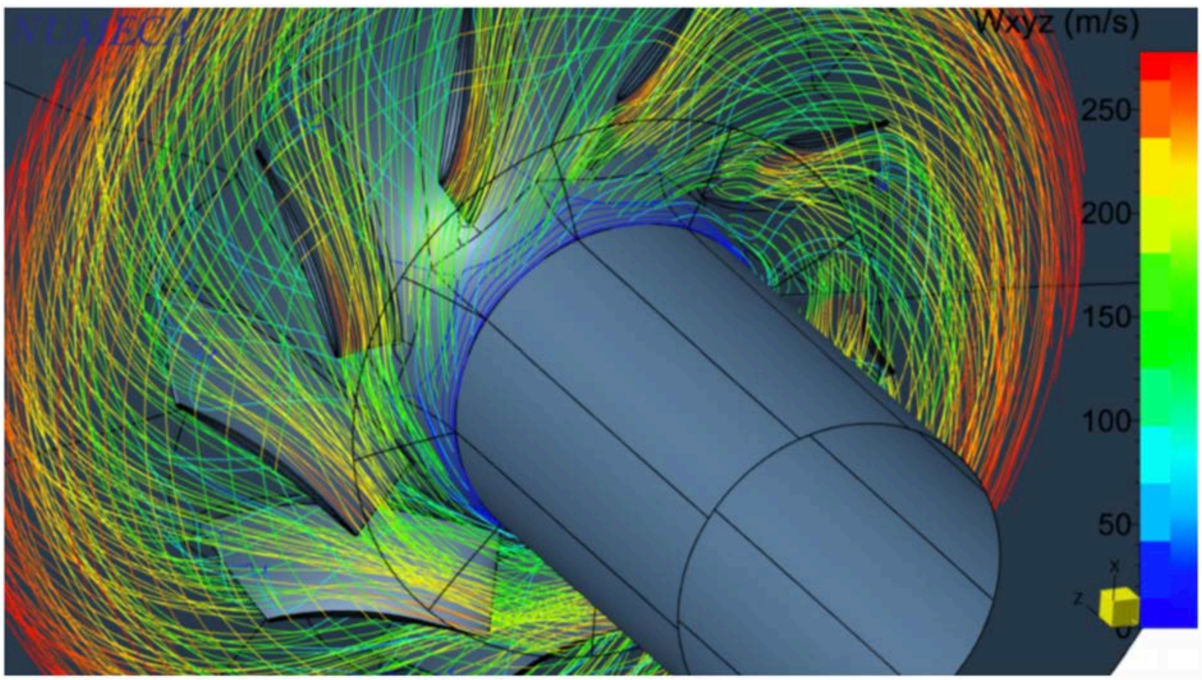


Figure 3-29 Example CFD analysis performed for an aero-rotor.

Prerequisite to the stated efficiency considerations, the performance envelope and characteristics of the aero-rotor impeller are critical machine design constraints. The simplest expression of this is establishing the expected pressure ratio over the altitude range. These will provide the basic conditions for machine design noted previously, as well as expected behavior (see "Rotordynamics (Spinning Masses)" on page 91) and performance (mass flow). While relatively straightforward calculations, failure to consider these may result in complete non-function of the air ballast moving system or unexpected behaviors, for example, oscillations during compressor surge at high altitude. Sufficient margin on these design factors will also account for manufacturing and environmental variations.

In addition to aerodynamic efficiency, similar electrical efficiency requirements exist. The electric motor design should also have the same amount of analytical rigor to avoid performance losses and instability. At upper operational limits dictated by the machine's rotordynamics (60,000-65,000 RPM), there are two additional unique challenges–both the motor and the motor controller.

From a motor perspective, special care must be taken in the design of the magnetics to reduce core losses in the motor. Also, mechanical construction of the motor is critical to be able to survive the loading conditions of high speed, as well as the large temperature swings. It proved extremely useful to include an RTD temperature sensor embedded in the stator windings to monitor in-flight temperature. This data can be especially critical for the prognostics and failure diagnosis techniques discussed below.

For the motor controller, efficient operation at these speeds is not trivial. Loon's motor did not include any sensors (hall or encoder) for motor position, as there was concern about the reliability and integration of these parts. This forced the motor control algorithm to run sensorless, which can be tricky for high speed motors, as sampling of the back EMF to determine position must be quite fast. Although slightly less efficient, all production systems used a trapezoidal drive waveform, but experimentation was in process to switch to sinusoidal/field oriented control to try to improve efficiency. Lastly, special care must be taken to tune the spin up of the motor, and time the spin up correctly with the opening of the valve, as this will apply a disturbance to the control system. Additionally, the temperature (and therefore friction) of the motor will change as it heats up, affecting the dynamics of the system (see compromises in "Bearings and Lubrication" on page 152).

Rotordynamics (Spinning Masses)

Given the nature of the turbomachines used by Loon, which run over a wide range of speeds due to different ambient conditions (ambient pressure, pressure ratio between balloon and ambient, and commanded power), it was critical to analyze the rotordynamics of the compressor very early in the design process. Loon's ACS required an accurate understanding of vibration modal behavior, as running through, or at, modal frequencies resulted in highly amplified responses leading to random failures. Consideration of temperature and its effect on modal response also proved important, as the damping response of cold O-rings, dimensional gaps, and greases changed significantly for both the rotational and translational response of the spinning masses.

Machine Learned Sensorless Prognostics

Ground-based, high speed, rotating equipment (especially in a capital equipment industrial setting) is often inspected on a regular basis, and/or has preventive service performed to ensure continuous operation. Given the nature of Loon's flight vehicle being airborne from launch to end-of-life, all of Loon's systems are necessarily non-serviceable. A predictive method for monitoring any critical systems (especially ACS) provided valuable steering and maneuvering time to the flight engineering team when planning flight paths and landing locations. Regarding the ACS, Loon attempted multi-pronged approaches to extend the flight vehicle service life duration. While the reliability section details the design-side implementations, including parallel compressor redundancy, this section discusses Loon's attempt at in-flight ACS life prediction.

As an initial attempt, Loon used a simple 3 node k-fold machine learning model with multiple sensor inputs, training it against known good (long lived) turbomachines as well as later life failures.

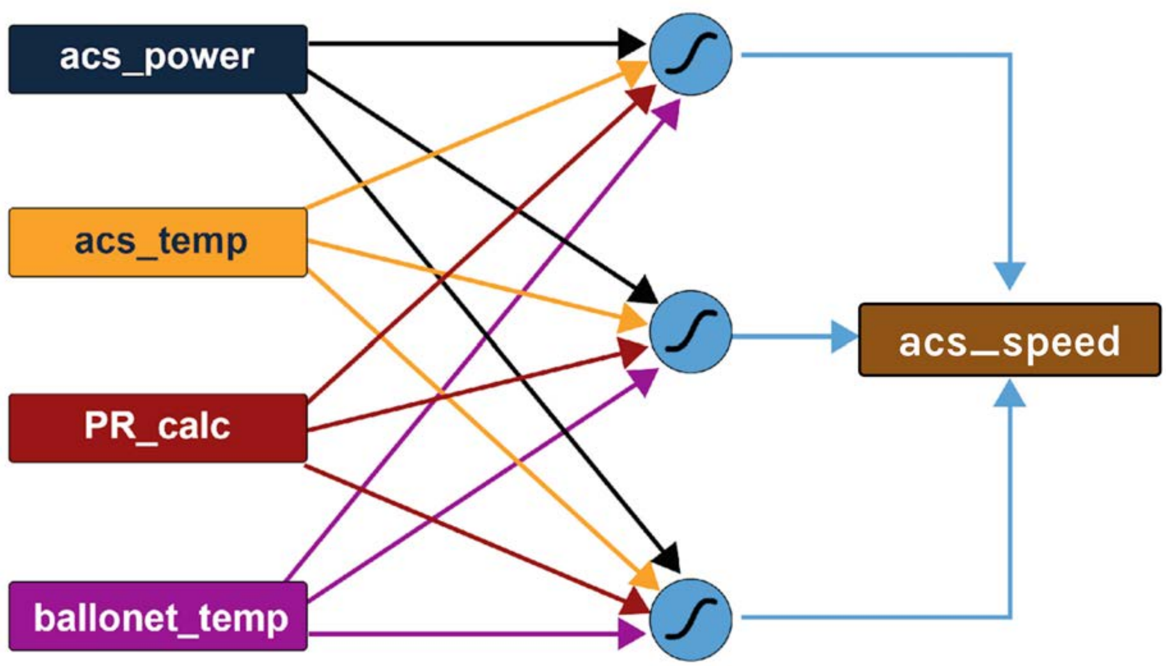


Figure 3-30 Three node K-Fold model diagram (k=5) for in-flight prognostics.

This model was simple enough to run locally on the bus at a relatively high rate, and the system could be monitored against deviation from the normal speed output of the model to determine if a malfunction was imminent. Some care needed to be taken in sampling, especially given automation-driven events, as some local transient conditions caused deviation from the model (for example, a system starting at a very cold overnight temperature). Additional work could also be done from the cross-validation standpoint to improve response to new data points.

Loon further refined this methodology by customizing the model on a per-compressor basis using an initial training process dataset for the first few operations of the compressor (including the in-flight "run-in" period). This allowed a much finer resolved known good speed for that specific compressor and accommodated the nuances of that unit's manufacturing (example, the amount of grease fill, motor variation or dimensional differences). This higher overhead algorithm necessarily ran from telemetry piped into a data center, where the data could be analyzed and the fleet health could be tracked via a dashboard.

In both cases, Loon found that if done over a significant dataset with appropriate model validation, systems can be accurately monitored for possible failures before runaway conditions, even without challenging direct sensors by combining the noted surrounding temperature, pressure differentials, and other system data.

One major shortcoming in Loon's use of both of these approaches is that they typically identify only one specific failure mode (long term fatigue damage to bearing races). This prevented early-life failure modes from being identified with these techniques. A considerably more advanced implementation could take advantage of specific characteristics of each failure mode, especially in the noted transient conditions across a more diverse sensor input array (for example, an electrical failure might result in a steeper drop in RPMs).

Reliability and Redundancy Approaches for Ballast Systems

The ACS was the only way to steer the flight vehicle sufficiently for viable navigation (Seahorse could enhance but not replace altitude control steering). As such, if it failed, the flight vehicle was no longer viable for its mission and could drift into areas with no overflight agreement. Achieving 100% full redundancy of the ACS, would be a poor system trade given the mass implications. In an ideal vehicle, an established agility requirement for a mission would be fulfilled by some number of compressor-valve systems as part of the ACS, where failure of one or more would still allow a mission to be carried out. This would be analogous to a multi-engine aircraft approach, where a performance envelope is established, and single or multiple failures would appropriately change the vehicle's behavior and agility in a defined but tolerable way.

Additionally, full redundancy requires the compressor-valve subsystems of the ACS would ideally fail safe (with an identified desired state–such as valves closed versus valves open) while also offering full operational performance. In the case of compressor-valve mass flow redundancy, the systems would be sized such that a failure would still allow for mission capability. This is a very stringent design challenge for the application in light of other design considerations discussed and would need to be determined as a systems engineering-level trade for a future flight vehicle. The net degradation of a failure versus machine size/mass/power would need to be evaluated for the appropriate sizing.

As an intermediate step Loon was pursuing a two compressor ACS running in parallel, providing a limited steering functionality should one compressor fail. Limited steering could be sufficient for certain missions but for high performance requirement missions or service areas the altitude rate of change agility was severely reduced by the loss of one of the two compressors. However, the important advantage of this parallel system meant that a single failure would still allow for navigation capabilities sufficient to avoid areas without overflight agreements and reach a favorable landing and recovery area. In addition, it meant that compressors operating life could be extended with significantly less risk associated with a failure.

Summary of Key Learnings and Innovations

- A complementary strategy for design and reliability is critical for achieving expected performance and longevity requirements.
- Ignoring rotordynamics behavior of wide operating speed range machines will lead to random failures.
- CTE compensation and consideration is needed to accommodate the dimensional mismatches from uneven heat dissipation for critical mechanical and electronic components throughout the dynamic ACS across the wide range of thermal conditions in the stratosphere (see "Passive Thermally-Compensated Bearing Preload Systems" on page 88).
- Bearing life analysis is needed to understand the inherent fatigue life of a compressor's bearing system–Loon only understood this late in the design process. Ideally, reliability specifications would have been used to select both the bearing elements and arrangement to achieve flight vehicle life targets.

- As flight critical, an ACS should have a “surplus” reliability, accommodating random and early life failure modes based on reliability parameters established (ideally tested or at least modeled with confidence intervals). As noted in the reliability guidance section a reliability budget and model should reflect this as well as the operational profile desired (full operational redundancy or limited agility redundancy).
- HASS and ORT should be a fully integrated process for critical systems like ACS. This was not yet implemented as extensively as we would have liked by the time Loon wound down (limited sample size ORT). These types of activities significantly reduce quality escapes and flight vehicle loss of mission viability (cost and operational risk). Ideally, the amount of margin in the design is well understood and allows for effective HASS, as well as establishing a metric during and post test (see "End-of-Line Spectral Fingerprint of Vibration at Time Zero" on page 89).
- An initial surplus of sensors on the machine and instrumentation are informative over the course of time to understand unknown behaviors and for diagnosis of in-flight failures with no known root cause; removing them early has potentially far-reaching consequences. Often, these can be low sample rate but accurate temperature sensors (See "Thermal" on page 152).
- Loon used a Lift Gas Concentration Sensor (with temperature and pressure auxiliaries) to determine if there is helium leaking into the air ballast chamber from the lift gas chamber. This was very valuable to understand both balloon and ACS health:
    - » Loon used a low-cost, off-the-shelf thermal conductivity sensor and calibrated it under flight conditions in a chamber with known helium concentrations to provide (very rough) estimates in flight.
    - » Non-trivial characterization and software filters were necessary to use this sensor effectively over long duration to estimate balloon life and leaks.
    - » Using this device in temperatures significantly out of specification led to sensor failures (both transient and permanent) in flight. Redundancy of this sensor could have been implemented in future vehicles.

Avionics, Bus and Power Systems

The flight vehicle's control avionics including power generation and storage systems were located on the bus suspended below the balloon by the mechanical downconnection. The power system evolved numerous times in terms of capacity as the power requirements of the service payload evolved and new systems such as Seahorse were added.

Loon's avionics system was largely driven by a few key functional goals:

- Supporting an increasingly large service payload with aggressive power, azimuth pointing, and stability requirements.
- Providing reliable and accurate sensor data to the data center-based modeling tools to determine state and health of the system, especially the balloon.
- Maintaining a high level of safety and compliance with air traffic control and regulatory body requirements for operation in many different air spaces.

This section covers a number of the key systems related to avionics and the power system as well as some key design decisions and learnings for those systems.

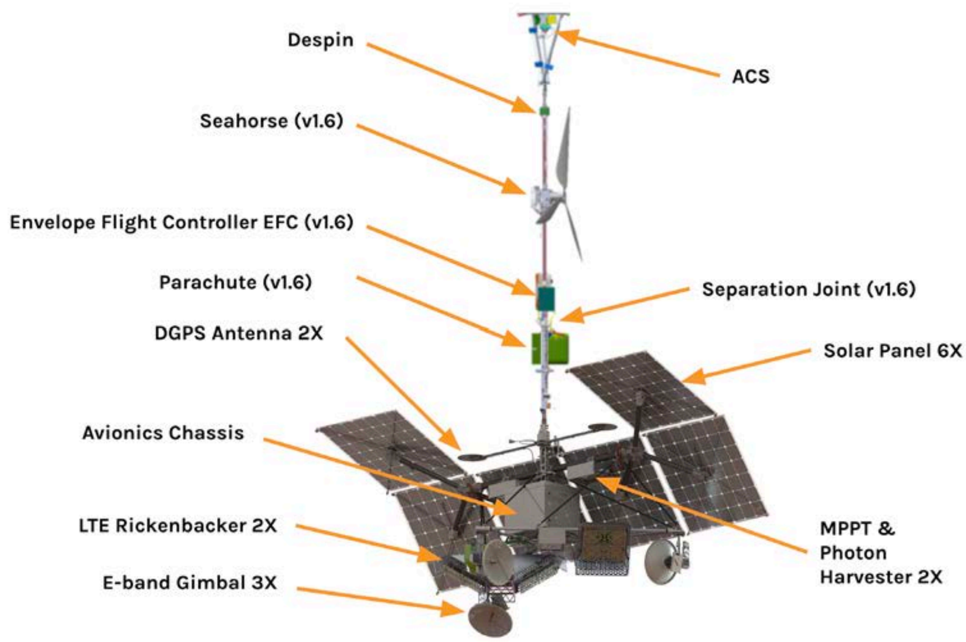


Figure 3-31 ***v1.4 bus with v1.6 additional components.***

Bus Mechanical Structure

The mechanical structure provided the mounting for all the components but also had to be robust through periods of high dynamic loading such as pre-flight handling, launch, and the extremely rare but possible balloon burst condition. In addition to this it needed to deal with very large temperature swings and gradients brought about in the cold stratosphere with bright sun causing unequal heating of components and UV degradation of materials.

The bus is connected to the balloon via the aluminum downconnection structural tube and a flexible knuckle, which allowed the balloon to tilt without inducing a tilt to the service payload. The long downconnection prevented unwanted contact between the bus or its hosted systems and the balloon. This distance also reduced shading of the solar panels by the balloon.

Power and other requirements evolved for several years so a common structural spine with an upgradeable service payload truss and solar structure helped accommodate changes in solar panels, batteries, and communications payload components without re-architecting the primary load bearing approach. The structure was primarily made from machined aluminum and carbon fiber tubing bonded to a nylon plastic using an epoxy adhesive.

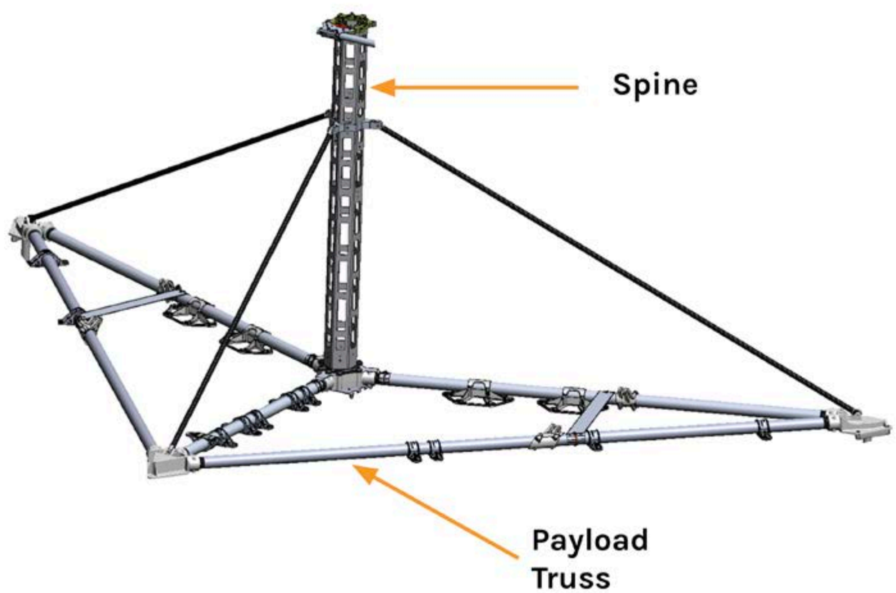


Figure 3-32 ***Basic bus and payload load-bearing structure.***

Power System

Power was a fundamental concern for Loon. While solar power was reliable since there were no clouds to alter solar collection from day to day, there were considerable draws on the power system. First was that the two systems that took the most power, steering with ACS, and powering communications equipment on the service payload, needed to be maximized for fundamental aspects of the Loon business model. Being able to steer well, i.e., to stay in the area desired, meant a reduction in the number of flight vehicles needed. Similarly, the greater capacity, range and on-time of the service payload, the higher the service delivered. Both took very significant power. Since solar power was only available during the day there needed to be significant power storage at night (batteries) and there were additional heating requirements to keep components from getting too cold.

Table 3-2 Comparison of power generation and storage by bus version

	v1.3	*v1.4*	*v1.6*
Solar	Four 49 cell Solar Panels ~3.3 m^2 900 W peak	Six 49 cell Solar Panels ~5.0 m^2 1350 W peak	Six 49 cell Solar Panels ~5.0 m^2 1350 W peak
Batteries	8-10 Hydra Battery Packs 2.4-3.0 kWh	11-12 Hydra Battery Packs 3.3-3.6 kWh	12 Hydra Battery Packs 3.6 kWh

Other than the solar panels, the rest of the power system was contained within the avionics chassis mounted directly to the bus spine. In the early days of Loon, the power system was quite simple and small–largely because there were little to no service payload power requirements. As Loon added and scaled the communications system, it required significantly more power. This requirement not only increases the size of the power system, but also the vehicle as a whole to carry the extra mass–which then increases the power requirements for navigation.

Because of this cyclic relationship, it is very important to keep in mind the relationship between bus and payload power requirements and power system size and mass, and to improve efficiency wherever possible. Some key optimizations are:

- Energy storage density (watt-hours/kilogram) of the battery
- Energy production density (watts/kilogram) of the solar panels
- Efficiency of the navigation systems

The next sections will dive into Loon's power system and how Loon chose to implement the batteries and solar.

Batteries

Currently, Lithium-Ion battery technology leads the market in terms of energy density. There are generally two options for packaging of these cells: cylindrical (encased in metal) and pouch (encased in a flexible shell). Pouch cells tend to have a slightly higher (~10%) energy density; however, Loon explicitly chose cylindrical over pouch cells as this allowed for much more robust (and lighter) mechanical design with minimal packaging complexity. Loon used the 18650 form factor cells but had plans to switch to 21700 form factor cells in the near future to take advantage of the higher energy density and future improvements to the cell lineups.

Loon also chose to use normal temperature range cells, as opposed to cells that can operate in a wider range (specifically colder) of temperatures. Cells that allow for colder operating conditions come with a lower energy capacity/density, which, with the payload and insulation strategy, actually cost more total energy than it took to keep normal cells in their operating range through heating. We found the optimal temperature for the cells was about 15°C (specifically–all cells should remain above 10°C, which is approximately the temperature at which their internal resistance increases). It is important to monitor temperatures at multiple points in the pack, as depending on the pack configuration, temperatures may vary significantly. For example, at times we saw a 5-10°C difference between the top and the bottom of the packs due to insulation configuration and heat sources within the chassis.

Loon's batteries were designed in a modular system enabling up to twelve 24-cell packs (12 cells in series, 2 of those in parallel) into a chassis, depending on mission.

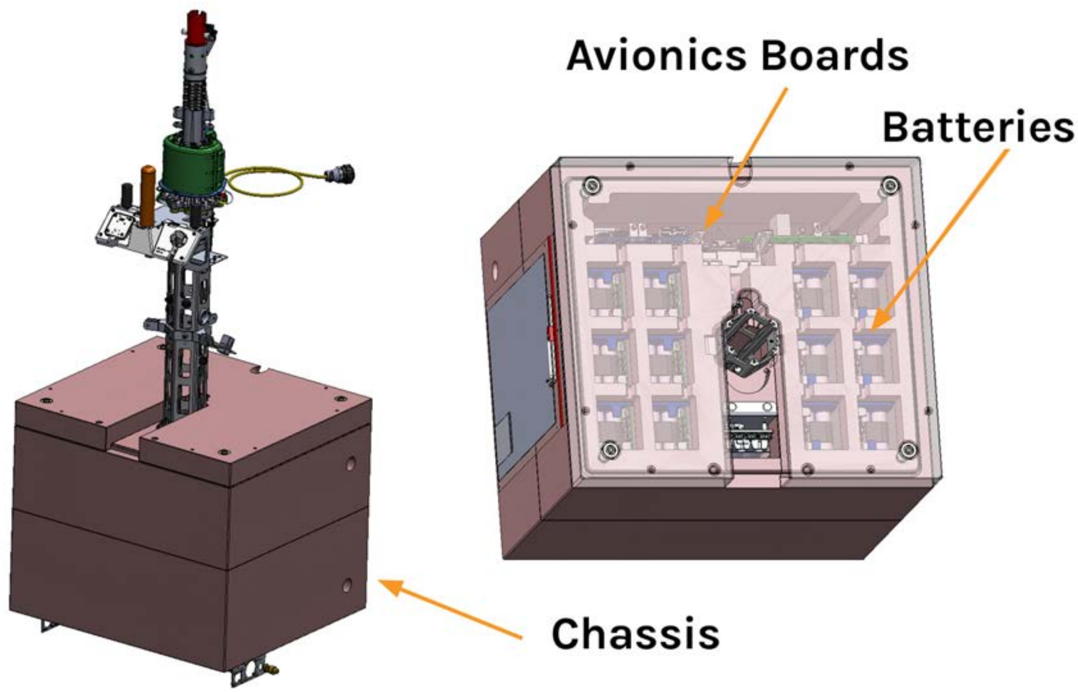


Figure 3-33 Avionics chassis.

Batteries were combined at a power distribution board and split into two unidirectional busses using ideal diode controllers to form a separate charge (solar) and discharge (everything else) bus. This worked quite well, as it allowed batteries of different charge levels to be plugged in without rapid charge transfer between packs. The downside of this approach, however, was that if any energy was accidentally pushed into the discharge bus (like by unintentionally braking a motor), it was easy to cause an overvoltage of other systems on the bus.

Solar

Loon considered and was monitoring the market for a number of different solar cell technologies, as the solar industry is ever evolving and, depending on system design, the tradeoff may go towards a different type of cell. There were three primary metrics used to evaluate the solar panels:

- Area Efficiency : watts per m^2 (maximize)
- Mass Efficiency : grams per watt (minimize)
- Cost Efficiency : dollars per watt (minimize)

Generally, there were three categories of cells, which each had different advantages:

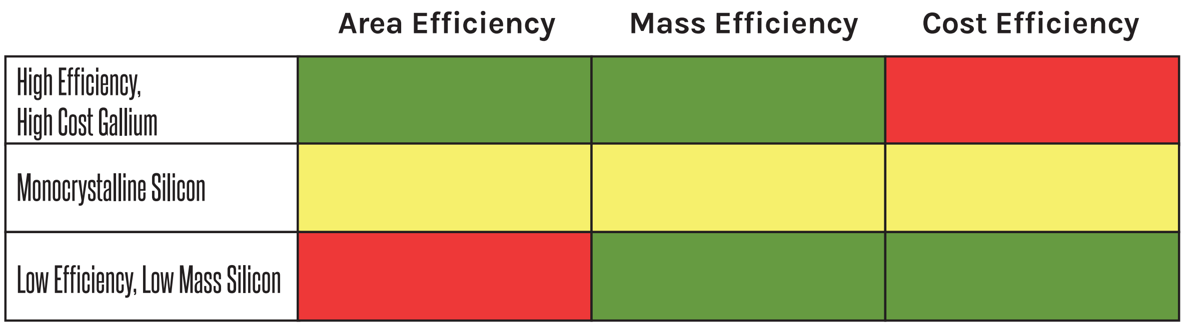


Figure 3-34 Solar panel category comparison.

Loon ultimately chose and remained with the monocrystalline cells, as they were generally good (but not great) at everything. Cells sourced were generally on the higher end of the efficiencies available, as this skewed the direction a little more towards the gallium cells, but without as much of a cost penalty. Overall, these panels/cells were found to be the right balance of cost efficiency and mass efficiency for the system without the need to grow the area drastically (which had logistical/structural mass challenges).

Loon also worked with the panel vendors to optimize the encapsulation of our cells for weight while also keeping enough rigidity such that the cells would not bend/crack in handling before flight and during any transient conditions during flight. An X frame system was developed and attached in the corners between the cells such that the panels were fairly well constrained at all times. This frame was attached at the panel vendor and the panels were EL and flash tested to ensure any unacceptable defects were identified before flight.

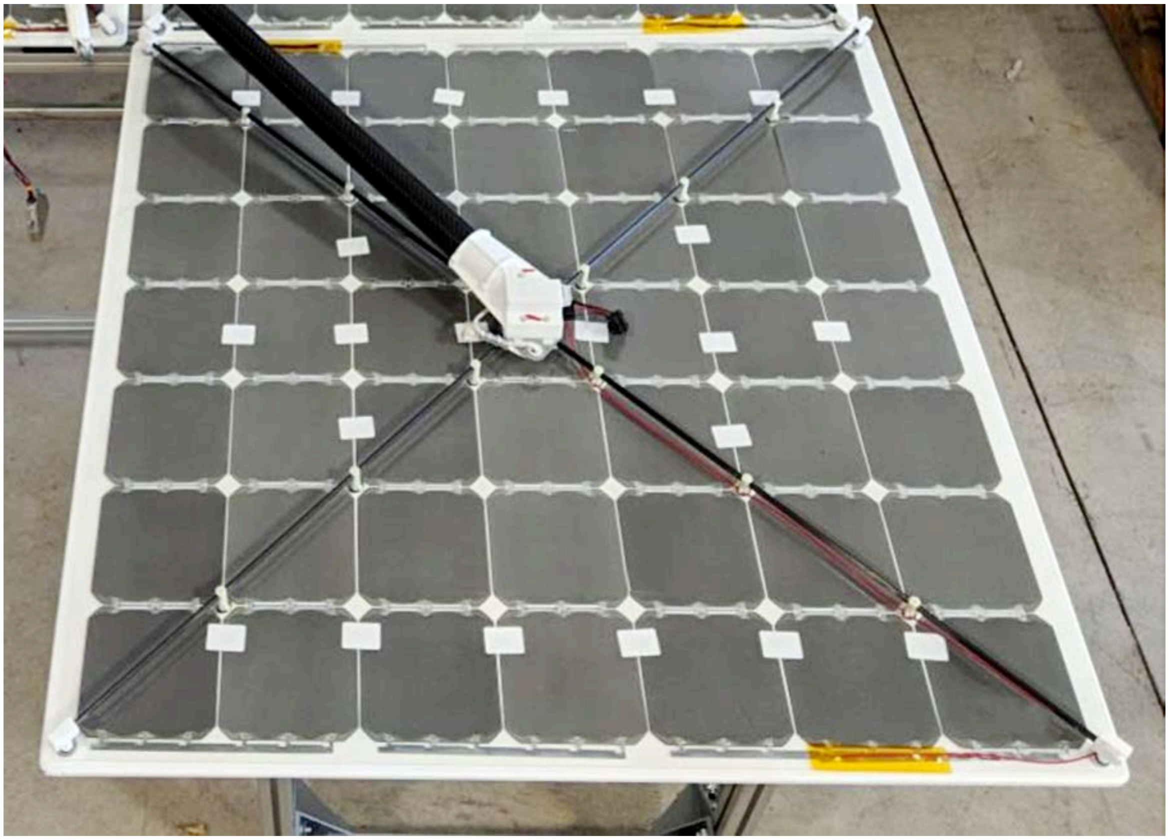


Figure 3-35 Solar panel (mounting side).

For the connection of the panels to the batteries, Loon chose to use maximum power point trackers (MPPT) rather than a direct charge method. This allowed the system to always maximize the power intake by the panels. While direct charging is less complex (and cheaper), it leaves some energy on the table, especially in the morning (when battery voltage is low), and increases the effective length of the night, therefore reducing the total system performance. There was also some self-imposed shading from the balloon at certain tilt conditions, so the MPPTs adapted better to these conditions.

Power Distribution Voltage

Loon chose a 12-series battery configuration which put the power distribution voltage in the 36-50.4V range. The goal was to maximize the voltage to reduce resistive losses in the cable while staying under the 50V handling threshold defined by OSHA requirements. This threshold was important because the batteries were placed into the chassis during flight prep at the launch site. If there are much larger loads (like in the case of an airship), it may make sense to increase the distribution voltage, as the size of the vehicle would also get bigger and therefore have longer cables where resistive losses become more significant.

Early on in Loon, power was distributed via a regulated 24V power line to most low power systems. Ultimately, Loon eliminated this lower voltage line, as it created extra power dissipation at the central regulator inside the chassis (where the heat was unhelpful) and reduced the overall system efficiency. The systems were redesigned to accept the full battery voltage at their inputs with positive results.

Azimuth Attitude Determination and Control

Since the solar panels were only pointed in a single direction on the bus, it was essential to rotate the bus to face the sun throughout the day, as the balloon naturally rotates due to internal convective forces and other ambient conditions. To do this, the flight vehicle has to both be able to spin the bus relative to the balloon, as well as determine where the sun is located. This system is also essential to maintain a constant heading for the service payload gimbals and LTE antennas.

Determination

In the early days, the bus relied simply on two photocells pointed about 90° from each other to determine roughly where the sun was in the sky for the purpose of pointing the solar panels at the sun. The bus simply searched for when radiation on the two sensors was equal, which meant the solar panels were pointing toward the sun. This was sufficient for solar (and we kept it on the flight vehicle as a backup) but did not provide enough precision for the communication systems (establishing balloon-to-balloon and balloon-to-ground links, as well as consistent pointing of sectorized LTE). To solve this issue, Loon added a Dual-Antenna GPS receiver which provided a high precision heading and position. Since this was available, Loon also used it for solar pointing.

Despin Rotating Actuator

To allow for the rotation of the bus, Loon designed a rotating mechanism, called despin, inline with the downconnection between the balloon and the bus/payload. This mechanism contained both an actuator for the rotation, as well as a slip ring to allow power and the CAN communication network to pass through without the need to “unwind”. Both of these elements created unique reliability challenges.

The initial implementation of the actuator used a small brushless motor and a multiple spur gear drive. This worked reasonably well, but was rather heavy, had some backlash, and was difficult and expensive to manufacture. Loon later migrated to a harmonic drive and stepper motor system that proved very reliable and was significantly cheaper and easier to manufacture. Low debris generation (to prevent binding failures) and lubrication selection (-90°C operation) were important aspects.

Loon also found the long-term operation of the slip rings to be a significant reliability challenge for the flight vehicles and was never fully resolved. Interestingly enough, we found the CAN signals that passed through to be more sensitive to early failure compared to the power, as failure first showed up as a significant AC impedance change over the mechanism. Through experimentation, we found gold-on-gold brushes/rings were these most reliable and increasing the diameter of the rings also improved results.

Early degradations of this system were detectable (through the presence of errors on the CAN bus) and typically were not immediately catastrophic. We developed a solid characterization and understanding of the time-to-failure and were able to prevent the mechanism from spinning (which stopped the errors) before complete failure to safely steer and land the flight vehicle.

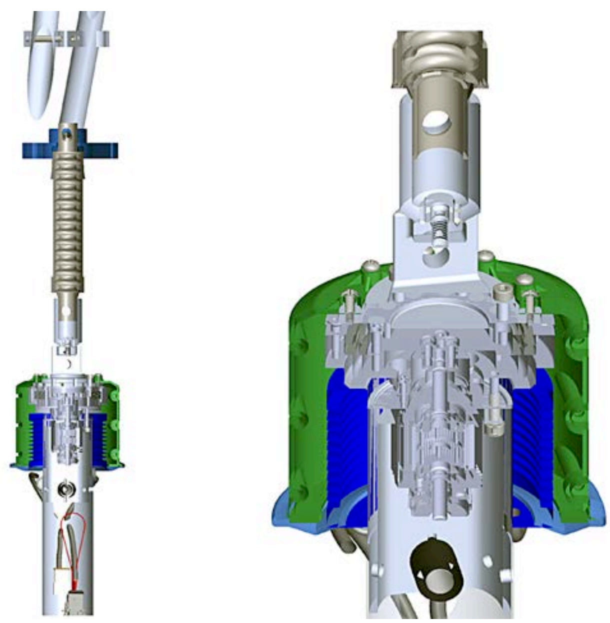


Figure 3-36 Despin rotating actuator.

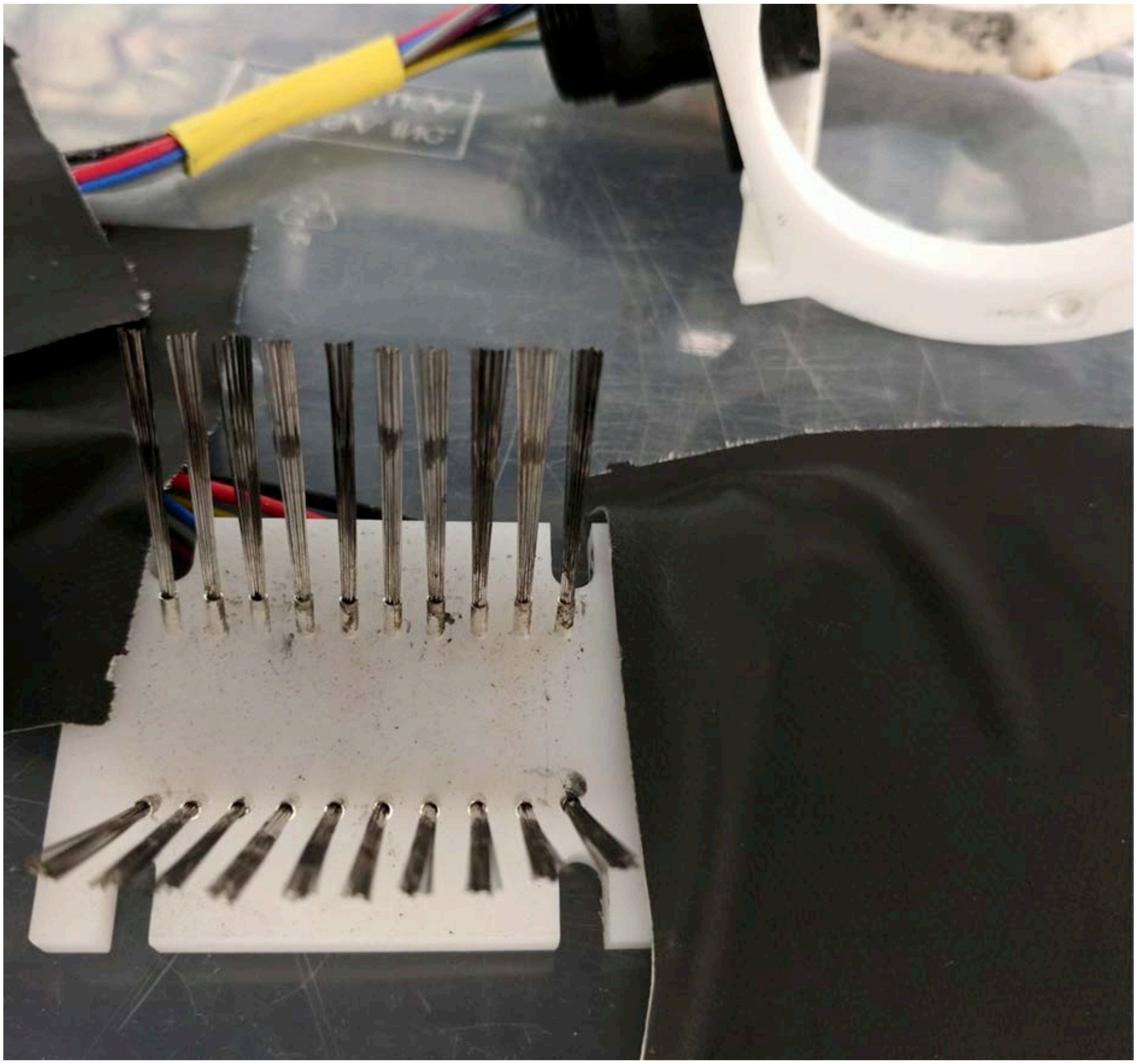


Figure 3-37 *Despin brushes.*

For future systems, Loon was exploring other possibilities for data, such as optical or RF, but would likely have stuck with a slip ring for power.

Powder Ballast for In-Flight Reducing of Flight Vehicle Total Mass

In addition to the air ballast used by the altitude control system, Loon found it useful to include a steel powder ballast system that could drop mass over the course of the flight vehicle lifetime to reduce the total system mass and account for any leaking helium, thus increasing the service lifetime of the flight vehicle.

Powder ballast could also be used to induce a rapid ascent to add altitude margin above a storm that is forming below the balloon and has a height close enough to the balloon to warrant increasing the balloon's altitude ceiling. See "Hardening Systems to Storm-Induced Electrical Activity" on page 154 for more on storms. In certain regions storms could develop quickly and cool the air above them such that the balloon would lose buoyancy and risk what's known as thermal runaway where the changes in buoyancy due to temperature makes the balloon descend. In instances like this being able to rise rapidly without requiring significant time for the ACS to vent air ballast was a very helpful contingency measure.

Loon developed two types of powder ballast systems: a fixed 5 kg system, as well as a variable drop system that went up to 25 kg. On early systems, we simply used the 5 kg systems, as they were fired with the same squibs cutters which were used elsewhere on the balloon for flight termination and parachute deployment and required minimal additional electronics and firmware. Squibs or squib cutters are activated by a small explosive charge when a voltage is applied. The specific device Loon used was widely used for parachute actuation applications.

Later, once the usefulness of this system was discovered, we designed a system that allowed for the dropping of custom amounts of powder ballast depending on the situation. This elegantly simple system used a magnet to retain the powder and a coil to cancel the magnetic field when we wanted to drop some of the ballast. It proved enormously reliable and a highly mass-efficient way of enabling powder ballast. We had originally intended to use a lidar sensor inside the reservoir to determine how much ballast was dropped, but it turned out to be very consistent and more accurate to simply characterize the time the coil was on to determine how much was dispensed.

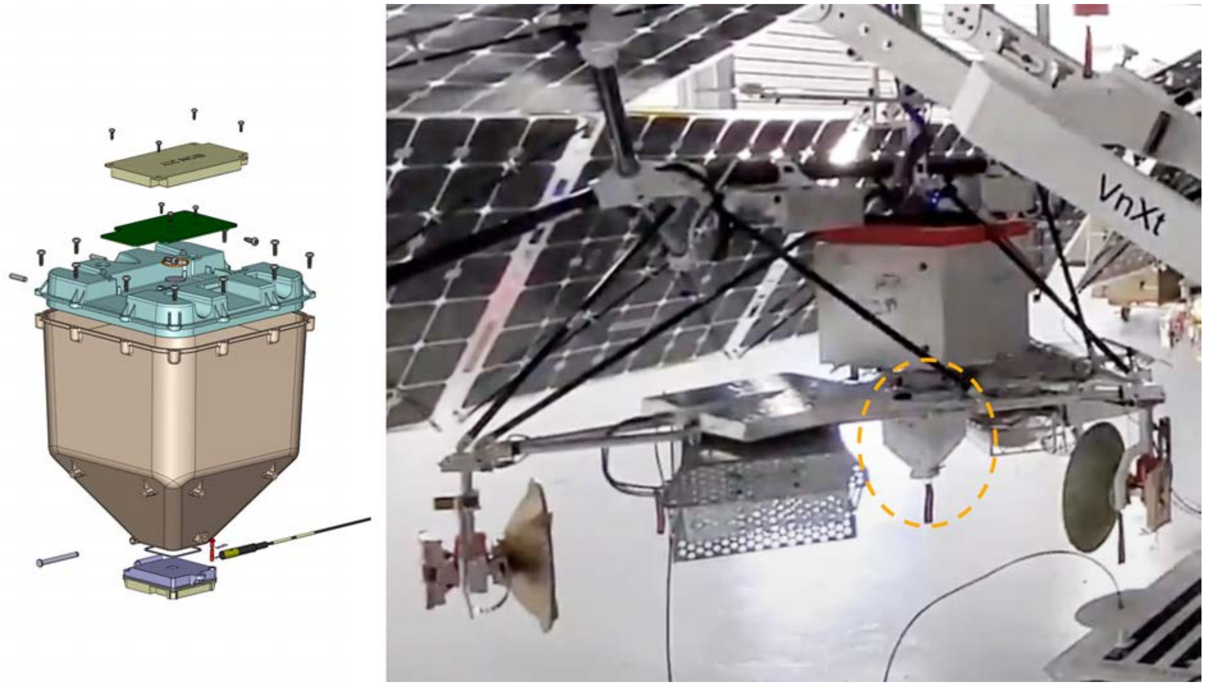


Figure 3-38 Variable drop-size powder ballast system.

Lateral Propulsion

Loon's Seahorse lateral propulsion system, which was deployed on the v1.6 flight vehicle, provided a second method of steering by inducing a small lateral force that moved the balloon approximately 1-3 m/s laterally (airspeed), depending on factors such as power, balloon size, and altitude, to help mitigate the effects of unfavorable winds, or assist in the transit to more favorable winds. In addition to the navigation benefit, the lateral propulsion cooled the balloon's lift gas (reducing daytime stress) through forced convection and enhanced its safety through enabling more precise landing locations.

This system was attached to the pumpkin balloons to improve their performance but primarily to gain experience with utilizing lateral propulsion for future, more powerful configurations on shaped, aerodynamic balloons. Given the drag of the pumpkin shape the resulting performance was modest. The following plot gives an idea of expected lateral speed vs. power for the largest Quail pumpkin balloon on the v1.6 flight vehicle at various altitudes.

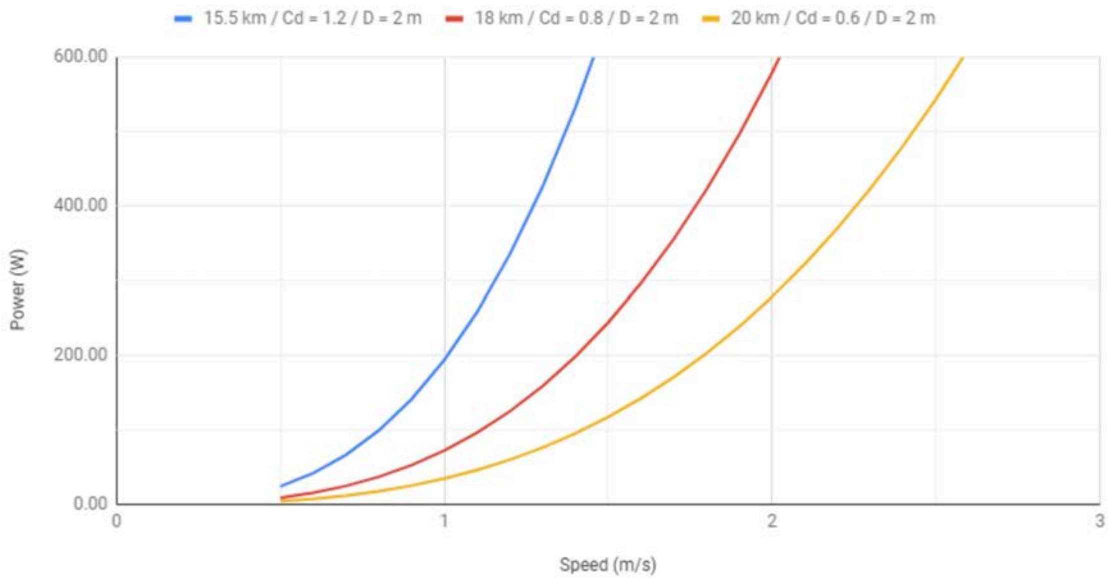


Figure 3-39 Lateral propulsion speed vs. power for Seahorse.

In this plot, observe that the power required is proportional to speed (velocity) cubed. This makes the first ~200 W quite impactful, but with drastically diminishing returns after that point. It is also worth noting that higher altitudes tend to be more efficient as the lower pressure leads to more aerodynamic efficiency, and the effective coefficient of drag is less, as the balloon tilts more at lower altitudes (due to more air in the air ballast chamber) leading to a greater frontal area (which is compensated for in the modeling by increasing Cd).

Loon built two iterations of the lateral propulsion system referred to as Seahorse Alpha and Seahorse Beta. The Alpha system was the first system designed to prove out that lateral propulsion was a worthwhile investment for Loon overall and was flown on customized v1.4 flight vehicles. The Beta version was a productionized version for v1.6 with a slightly larger propeller (2 m diameter vs. 1.5 m) and would be on all flight vehicles as Loon wound down.

The Seahorse systems consisted of a propeller sized and shaped for stratospheric air density mounted on the downconnection between the balloon and envelope. The system contained two motors—one to drive the propeller, and a separate motor which could orient the propeller to point in any direction required for steering independently from the orientation of the bus (which despin would keep pointing at the sun).

Key elements of the Seahorse system were:

- 2 m diameter propeller and hub
- Pointing axis and cable helix
- Drive motor
- Propeller brake
- Electronics (including custom motor drive)

Key learnings of each of these elements will be discussed in the following sections.

Figure 3-40 Seahorse Beta system on the v1.6 downconnection.

Propeller

For the Alpha (1.5 m) version of the propeller, Loon went with a single mold part with the hub integrated into the propeller. This worked quite well but had logistical (size of mold and shipping) challenges when scaling up to a larger propeller. So, when we transitioned to a 2 m diameter on Seahorse Beta, we decided to mold each blade separately and integrate them together with an aluminum hub. This solved some problems, but we quickly found the new prop significantly increased vibration in the system. This was a critical concern, as the E-band gimbals in the service payload needed minimal disturbances to maintain a strong mesh. Also, since Seahorse was mounted in the middle of the downconnection, vibration easily transmitted to other parts of the flight vehicle. When root-causing the vibration, we determined that even if the prop assembly was statically balanced, it was much more prone to dynamic balance issues with the multi-mold design, as it was possible for the blades to be slightly out of plane or have other asymmetries in the blades. Tightening control on the manufacturing process and slightly changing the design of the composite layup to improve stiffness significantly improved the overall system balance within tolerable levels. The new design's balance was quantified on the ground and correlated to the received signal strength indicator (RSSI) reporting from the gimbals, as well as in-flight accelerometer logs.

***Figure 3-41** Individual Seahorse Beta propeller blades and mounting hub in shipping packaging.*

In addition to the normal operation requirements, mechanically, the propellers needed to be strong enough to not break off during descent. Loon was particularly concerned about the prop getting entangled with the envelope film or tendons. Particular care was taken in material choice and testing to ensure the prop did not become detached with anticipated impact or entanglement.

Special attention should also be paid to the paint or coatings applied to the propeller, as many can be degraded by UV or flake off due to temperature cycling.

Pointing Axis and Cable Helix

The Seahorse Alpha system used a slip ring for data and power, much like the despin system. When designing the Beta version, Loon wanted to remove this from the design, as there were known reliability issues that we did not want to propagate further into the system. Therefore, we designed a cable helix which allowed for winding and unwinding of standard cabling and enabled 400 degrees of rotation for both a power and Ethernet cable running to the system. The system would unwind periodically (perhaps a handful of times a day in normal operation) but still be able to point in the direction needed without requiring the bus to rotate relative to the best angle for solar power collection. Through reliability testing, we learned that if a shield is needed on these cables, it is critical to use foil rather than a braided shield, as the braided shield will eventually wear and short to conductors in the cable.

Figure 3-42 Pointing axis cable helix.

For rotation of the system, Loon used a worm-gear based pointing mechanism connected to a stepper motor that allowed the assembly to rotate around the downconnection.

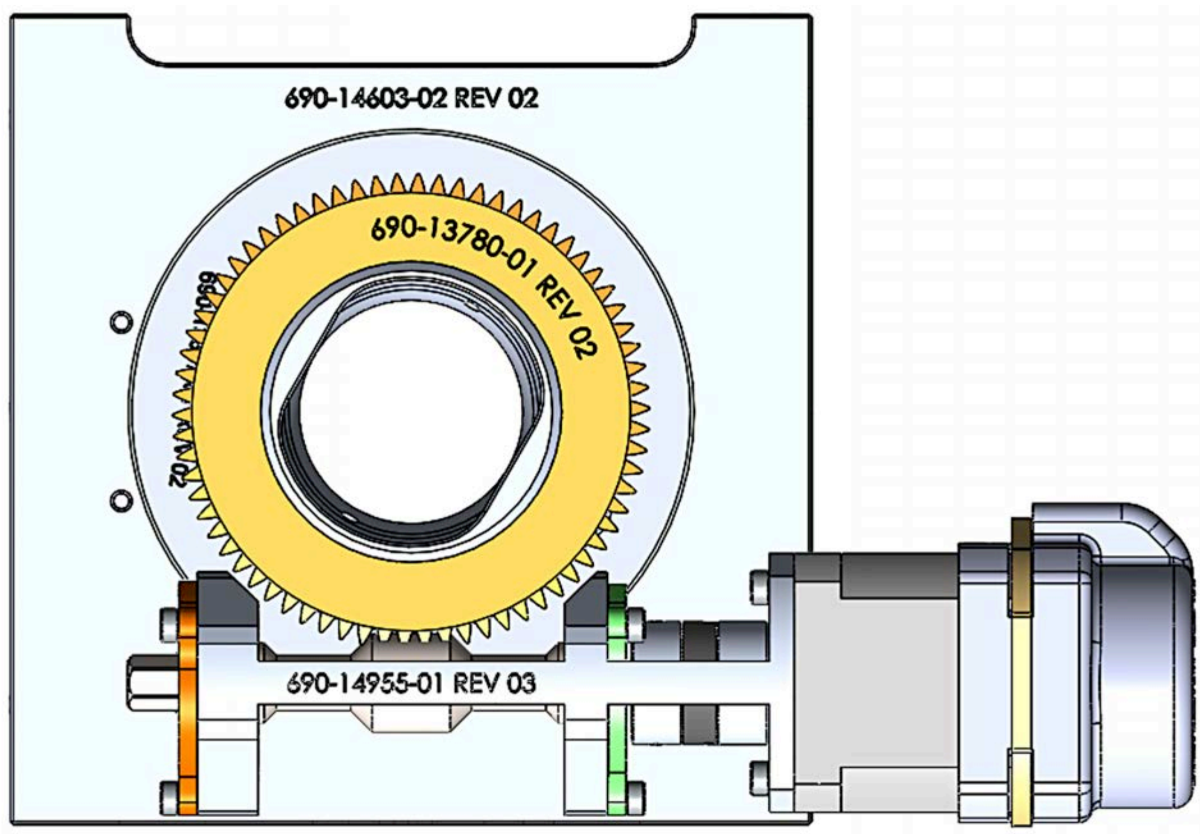


Figure 3-43 Pointing axis mechanism.

Drive Motor

For both Alpha and Beta systems, Loon used a brushless DC motor to drive the propeller. For Alpha, Loon was able to use a mostly off-the-shelf motor paired with a gearbox for the sake of rapid development. When designing the Beta system, we decided to partner with a vendor to custom-design a motor that would be a direct drive system, increasing the overall efficiency, as there were no gearbox losses. When doing system trades, this was beneficial, even with the marginal weight increase. We expected this to be the case for prop sizes up to the 2.5-3 m range, in which case the torque required is significant enough that the weight of the typical motor does not make sense compared to a gearbox, as well as the extra batteries and solar to compensate for efficiency loss.

It is worth noting that due to the wide range of altitude and power levels, the operating region of the motor is quite large, which makes optimizing for efficiency slightly challenging. This plot illustrates estimated motor torque, speed, and efficiency at different altitudes and power levels. Loon was able to increase efficiency by 2-3 percent (effectively shifting the most efficient band closer to the operating regime) with some changes to the motor but landed in the high eighties and low nineties for efficiency at most operating points.

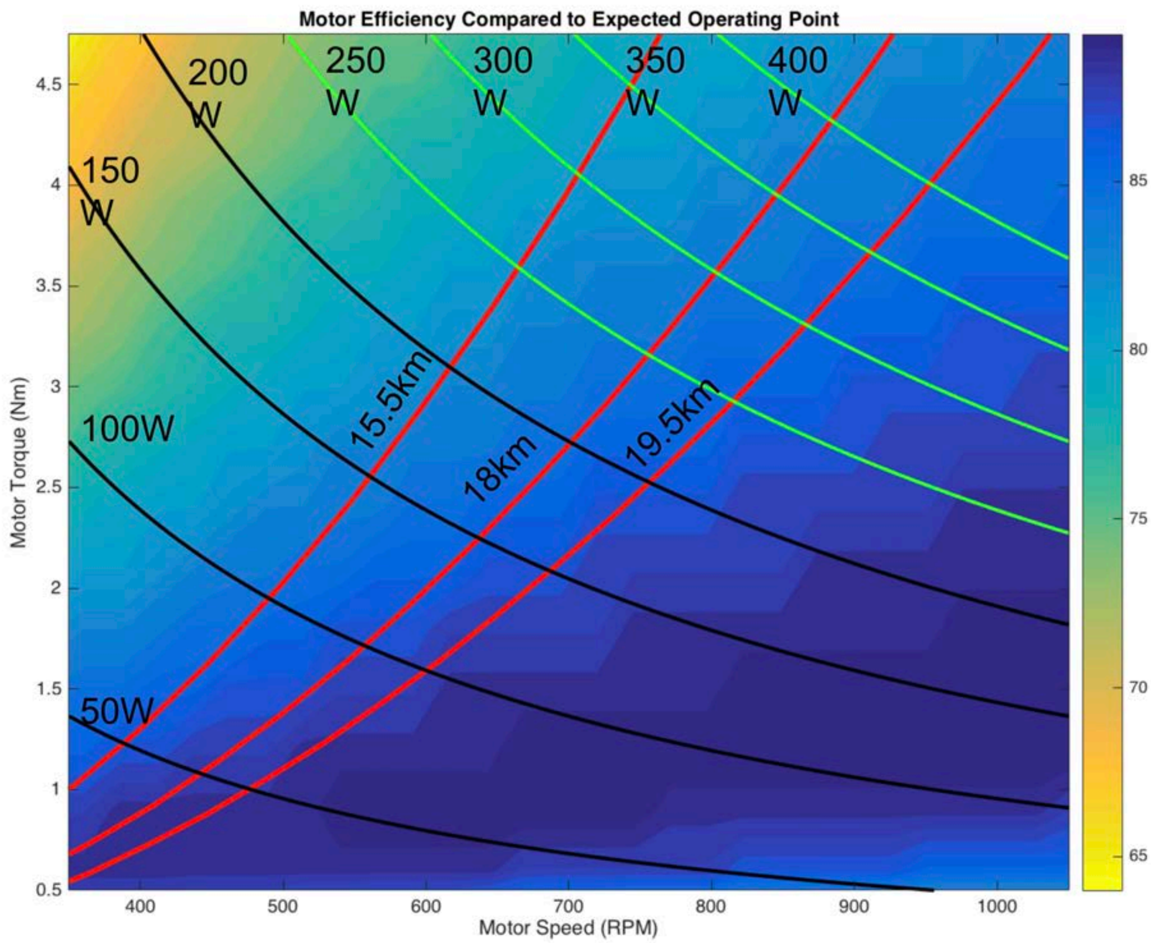


Figure 3-44 Motor efficiency plot by altitude and power level.

Propeller Brake

During development of Seahorse Beta, Loon came to the realization that during a period of high rate descent (which was nominal for the flight vehicle as part of the bus-separation descent system), the prop could spin up through auto-rotation, potentially to speeds greater than the normal operating speed which could lead to failure of the system, or overvoltage of the system bus due to the motor speed.

In finding a solution to the problem, Loon had a few key requirements:

- Brake must be active when no power is applied (so that if the system lost power during flight or descent, the brake would still be active).
- Brake must be able to stop a spinning prop (if an unintended termination were to occur).
- Brake must resist the highest dynamic pressure seen during the worst-case descent profiles.
- Ideally, the brake should be reusable, such that a mistake in triggering doesn't permanently render the system useless.

This led Loon to implement a wedge brake design, which used a spring to push a steel wedge against the aluminum propeller hub. The interaction of the surfaces leads to a galling which creates a high coefficient of friction. The wedge design meant that the higher the forces turning the propeller, the higher the reactionary force. We used a solenoid to retract the brake during propeller operation, and a holding magnet once retracted to reduce the amount of power required to keep the brake retracted.

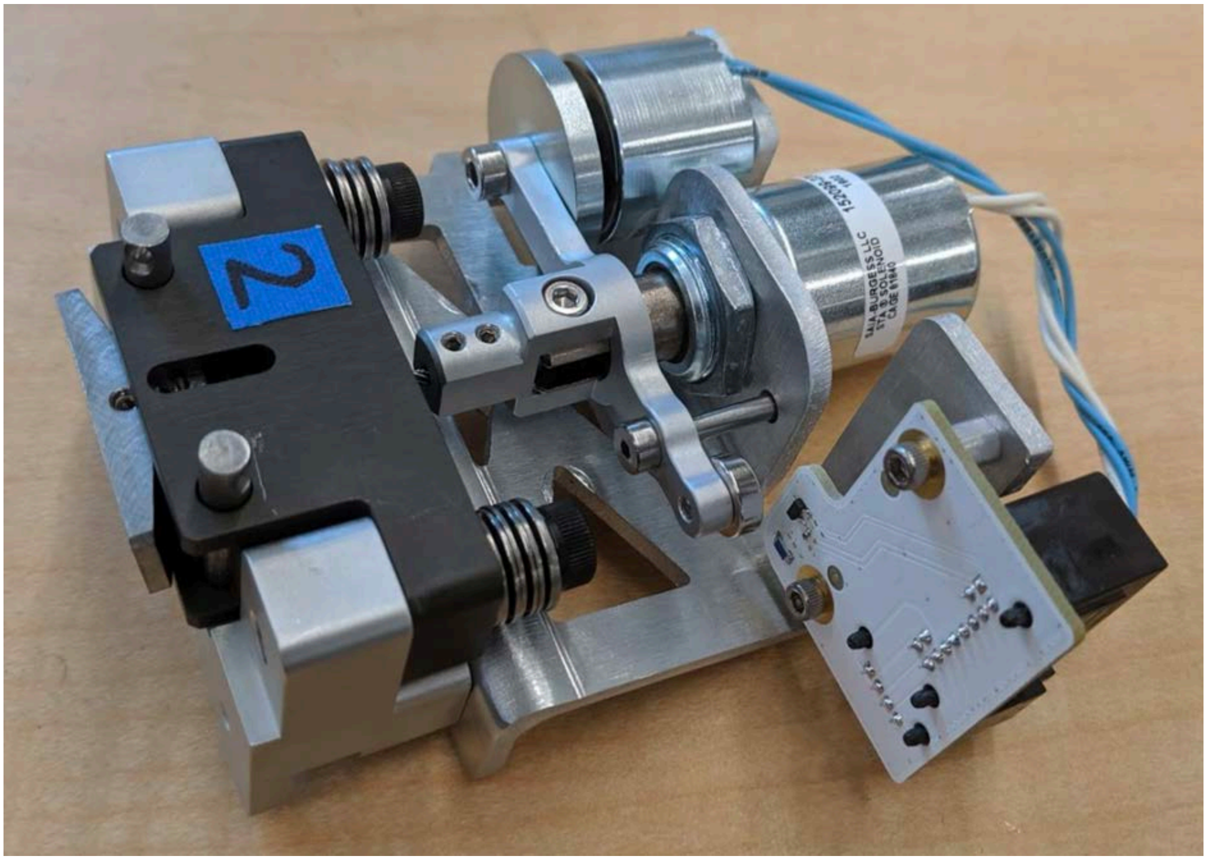


Figure 3-45 Propeller brake mechanism.

Electronics

Due to the complexity of the electromechanical assembly, the electronics to drive Seahorse also ended up being quite complex. Loon implemented a custom motor controller for the drive motor, which allowed optimization for high efficiency. We also implemented a custom warmup procedure that allowed us to preheat the drive motor to -60°C before significant movement to reduce wear on the bearings and increase life.

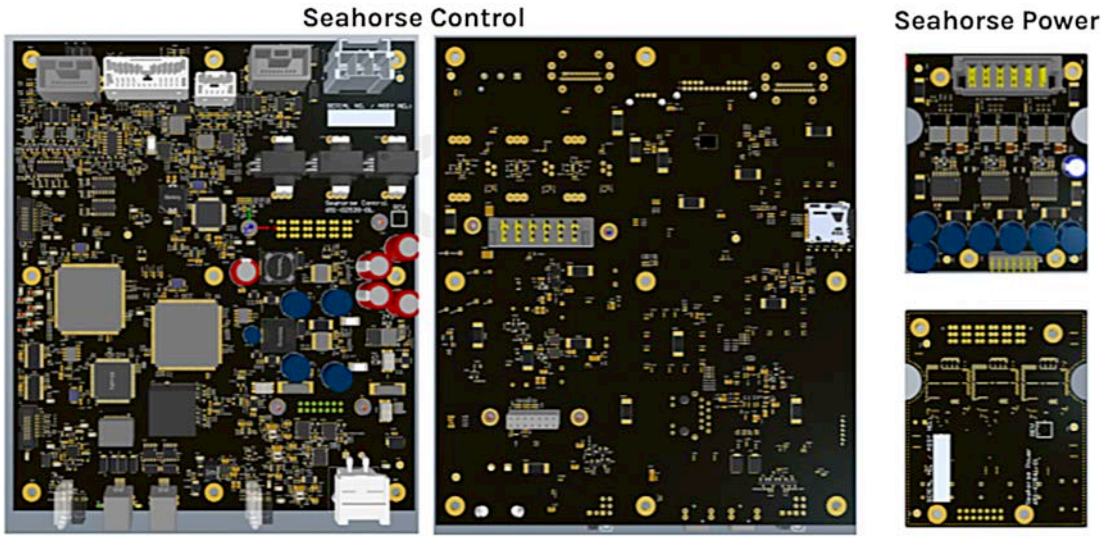


Figure 3-46 Seahorse control and power boards.

Flight Test Results

During flight testing a number of interesting effects were evident. When Seahorse increases the relative velocity of the flight vehicle by running the propeller, the change in pressure below the balloon descends the altitude of the flight vehicle by up to 200m. When thrust is applied in a location that is not the center-of-buoyancy of the balloon, the balloon also tilts which in turn changes the drag characteristics of the system.

One interesting note was that the drive electronics required far less heat dissipation than initially assumed, likely because when generating heat, they also generate some forced convection. In general, Loon saw an increased need for heating of the rest of the electronics in the system when the system was running during the daytime. The forced convection roughly eliminated the effect of radiation during the day (increasing the daytime heating requirements to roughly equal to the normal nighttime heating requirements). At night, however, running the prop seemed to have little effect on the heating requirements.

Since telemetry reporting from stratospheric systems is often quite slow (at best 1 point per 10 seconds), a suggestion for being able to more easily understand the flight test results is to implement an onboard calculation to convert the GPS course and speed over ground vector to a parallel and perpendicular component with reference to the direction of the propeller. This was very helpful, as Loon was more easily able to see the step change in speed in the parallel direction, and ideally saw no movement in the perpendicular direction (or if we did, the team knew there were other effects at play). Additionally, having the calculator onboard the balloon ensured that all input data was sampled at the same moment in time and overall required less telemetry bandwidth, letting us send the values more frequently.

Overall, Loon was generally able to confirm that the lateral propulsion system worked and produced roughly the speeds expected (with fairly large error bounds). The next steps for this system would be to better understand why, in some cases, the system did not move as predicted (one possible cause is the changing altitude effect caused a shift in wind direction) and better model and take advantage of this in the simulation and steering algorithms.

Future Designs

As mentioned before, one of the key motivations of developing the Seahorse systems was to learn more about lateral propulsion with the intent to use it on a shaped balloon/airship system, where lower drag can lead to a much more significant steering improvement (such as 6-10 m/s, rather than 1-3 m/s).

One of the most significant decisions Loon anticipated was the tradeoff between fewer larger propellers vs. more smaller propellers. The plot below shows one particular case of this.

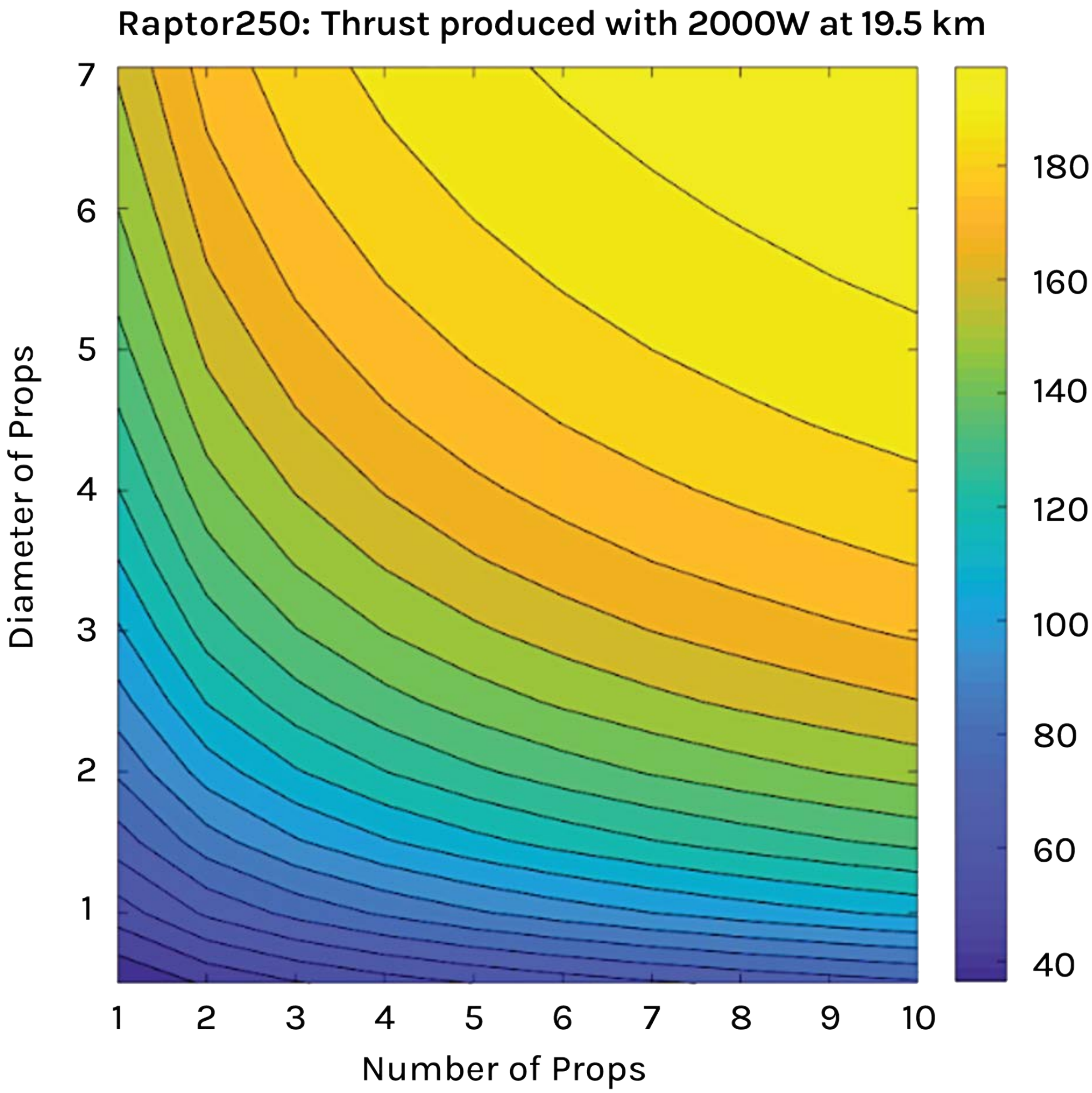


Figure 3-47 Comparison of thrust at 2000 W by number of props vs. diameter of prop.

Loon could get similar thrust out of four 2 m diameter props than with a single 4 m propeller. There are also very interesting trades on motor weight, efficiency with and without a gearbox, logistical and manufacturing challenges and reliability/redundancy. As Loon wound down the indications from analysis suggested that a sweet spot in terms of these trades might be 2-4 propellers in the 2-3 m diameter range.

Intra-Balloon Avionics Network

CAN was used extensively for most of Loon's Avionics networking from early on in the project. For the most part, it worked great and was quite flexible for adding and removing subsystems. As time went on, we experienced data rate limitations. We could only run up to 500 kBaud with the very long cable length which ran from the bus to the apex on top of the balloon and troubles with storm hardening also occurred, primarily because of the very long cable. We found Ethernet to be a good solution for both of these and started using 4-wire 100 Mbps Ethernet between new systems.

As the systems started to get more mature, Loon put effort into isolating the service-related communications and safety-critical avionics systems so as to reduce the likelihood of a fault within communications systems affecting a safety critical system. This was done by creating a software Ethernet firewall on the node connecting the two networks, as well as removing the CAN connection to all communication systems.

Mode S and ADS-B Out Transponder

Loon used the TT26 transponder as one of low size, weight, and power (SWAP) with certified Mode S and ADS-B (certification was necessary to build the safety case for overflights in many countries). Loon was also able to use the ADS-B outputs to track the balloons with sites like FlightRadar 24, as well as listen to other balloons' squawks from an ADS-B receiver for a lower latency estimation of balloon position when forming B2B E-band links. Since the development of TT26, another product emerged, the ping200x, which we also used successfully.

Pressure (Altitude) Sensor

While certified transponders typically have good, calibrated pressure sensors, Loon found early on that knowing the absolute ambient pressure of the balloon with extreme accuracy and precision was critical for estimating the balloon's health and state. There are not many good off-the-shelf sensors for the required altitude ranges with digital outputs, so Loon built its own using a MEMS analog sensor and a precision 24-bit ADC. We also developed a calibration procedure and ran each unit through this process to significantly enhance the quality of reading.

Satellite Communication

Loon used two global SATCOM networks to allow for redundant command, control, and telemetry data exchange to and from the data-center based fleet management system, these were the Iridium SBD and Inmarsat networks. These are fundamentally different networks: Iridium has a global constellation of many moving satellites in low-earth orbit (LEO) while Inmarsat has fewer in geo-stationary orbit (GEO), 40 times farther away than Iridium. Because Iridium is in LEO and always moving, it tends to be less sensitive to orientation of the antenna, so Loon took advantage of this for use in more unpredictable conditions, like the balloon descent. Inmarsat, on the other hand, allowed for larger packet sizes which was quite useful at float for configuring the communication systems. In both cases, care should be taken to minimize losses in the modem-to-antenna path, as well as choosing appropriate antennas with the correct gain pattern, taking into account region of operation and potential tilt of the antenna. Note that the experiences with these products was largely due to Loon's unique environment and challenges.

Table 3-3 Satellite communications systems used by Loon

	Iridium	*Inmarsat*
Sky -> Ground Latency	~5-20 seconds	~50-90 seconds, 3+ minutes worst case
Message Size	340 bytes (sky -> ground) 270 bytes (ground -> sky)	1022 bytes
Maximum Packet Rate, Typical Utilized Packet Rate	1 packet per minute	1 packet per minute max, 1 packet per 3 minutes utilized
Modem Location	Primary Flight Computer (Bus)	Satcom Bridge (Balloon top plate)

It is also worth noting the location of the modems. Loon always had Iridium located on the Primary Flight Computer (PFC) inside of the bus. This gave Loon a central location to control power and data throughout the vehicle. Inmarsat was originally also located on the truss outside the bus but was later moved to the top of the balloon with an added backup battery to provide backup power and communication to the apex to reduce the likelihood of an unintended cutdown due to CAN or CAN cable issues. This allowed for significant safety improvements as a command and control outage cutdown could be coordinated with air traffic control (ATC) and the location picked from the path overflown while on battery backup. When the bus-separation descent system was introduced, a second Iridium modem was also added to the Envelope Flight Controller (EFC) on the balloon, such that each of the two vehicles (bus and balloon after separation) had its own Iridium SATCOM connection.

Since data rates are so low over the chosen SATCOM channels, Loon developed a telemetry system that prioritizes certain data over other data to ensure that data needed every minute or urgently when it changes is sent first, then the rest of the packets can get filled in (round-robin style) with data that is less time sensitive but requested for monitoring the system state.

Neither system provided built-in security solutions sufficient for our use case, so Loon implemented its own authentication and, optionally, encryption. The command and control and telemetry messages did not contain sensitive data or trade secrets, so Loon opted to only sign messages for authentication, preventing attackers from spoofing or replaying the commands or telemetry, to save on the extra data that would have been required for full encryption.

In addition to SATCOM, Loon's own B2x E-band mesh network, supporting almost 1 Gbps, was used for high-rate telemetry when a flight vehicle was connected to the mesh.

Flight Termination and Descent Systems

A key element of the flight vehicle are the flight termination and descent systems. These subsystems provide the means to safely initiate landing and manage the descent of the balloon when needed at end-of-mission or as required during its lifetime for both planned and unplanned contingencies.

Flight Termination System (FTS)

The goal of the flight termination system (FTS) is to produce very reliable controlled venting of lift gas when commanded or automatically at the appropriate time in the case of loss of command and control. The system is located at the top of the balloon in the apex assembly and consists of two spring-loaded rotary cutters, triggered by squibs, which cut bolts to release the cutter arms. The cutter arms cut holes in a film window in the feature plate within the apex assembly. This allows the lift gas to vent up freely and causes the balloon to descend through loss of buoyancy. These components allowed for very high reliability.

The FTS system reliability is extremely important to avoid the condition of the balloon continuing to float once termination has been triggered. See "FTS Squib Triggering Logic" on page 145 for more details on the control system and triggering logic for the FTS.

Likewise, the open area and the rate of venting needs to be predictable to enable accurate descent trajectory simulation. See "Landing Zone Prediction Accuracy" on page 137 for more context.

The FTS system provides three cutting methods:

- Dual rotary cutters
  - Hard Terminate for primary termination
  - Soft Terminate for redundancy
- A single vertical plunger cutter
  - Super Soft Terminate for contingency landings with a very low landing speed

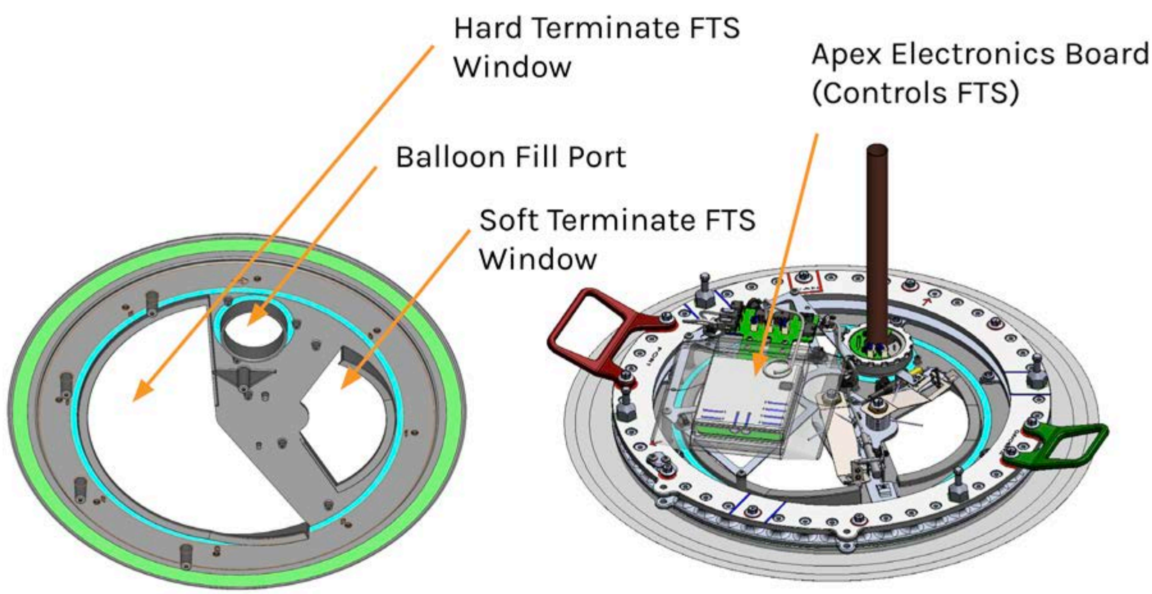


Figure 3-48 ***Hard and Soft Terminate cutter windows in the top apex assembly.*** Feature plate shown alone on the left.

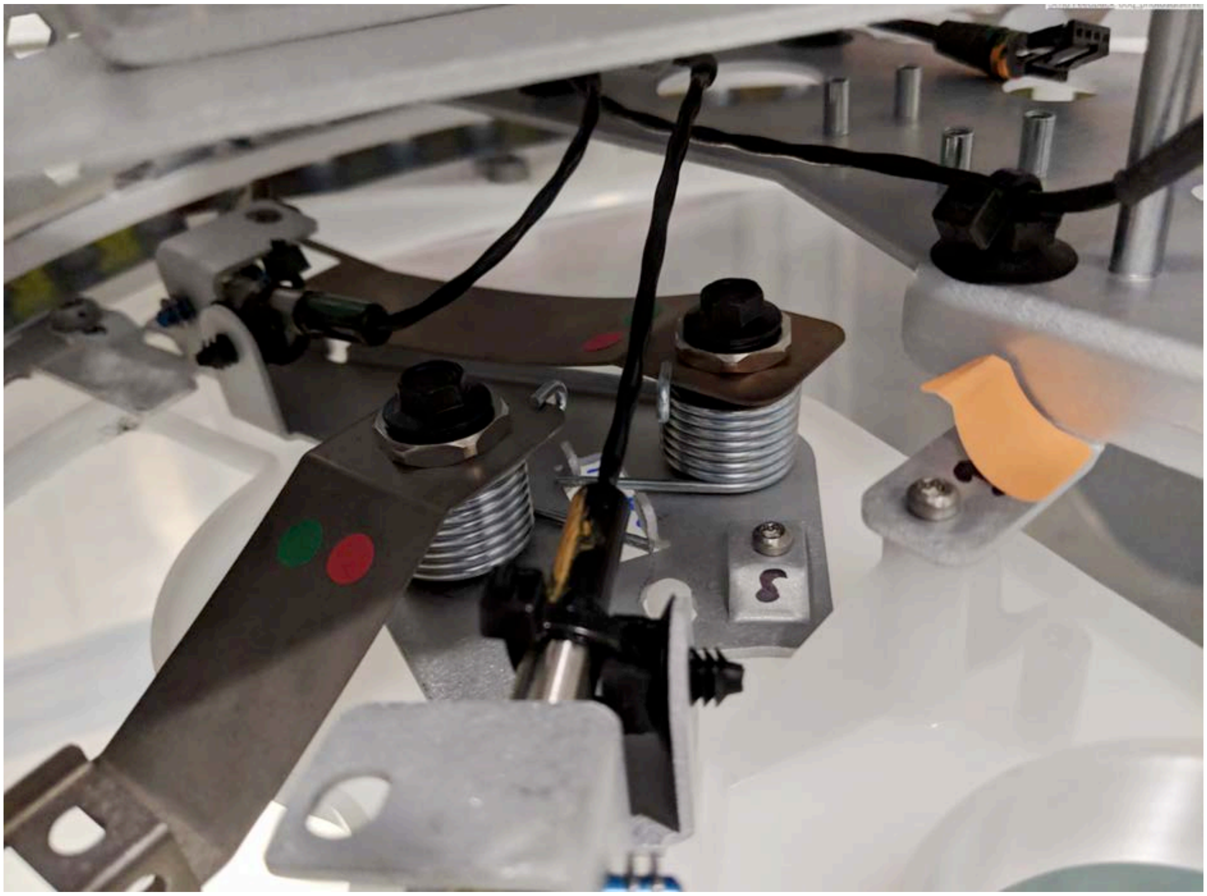


Figure 3-49 Hard and Soft Terminate rotary cutters.

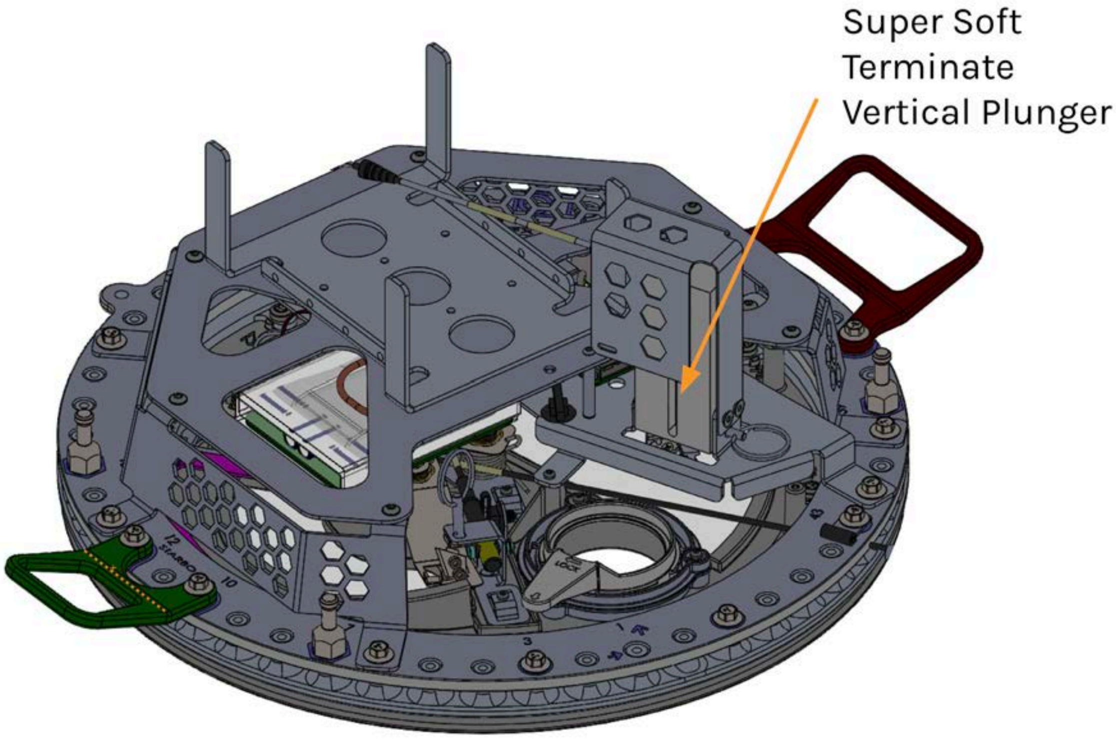


Figure 3-50 Super Soft Terminate vertical plunger cutter.

Descent and Parachute Systems

Until 2019, with the development of the new bus-separation descent system on v1.6, the parachute was deployed from the bus to float on a tether well above the flight vehicle and deploy the parachute out of the turbulent region in the balloon's wake. This kept the parachute from interacting with the envelope during the relatively low-speed descent generated by Hard Terminate.

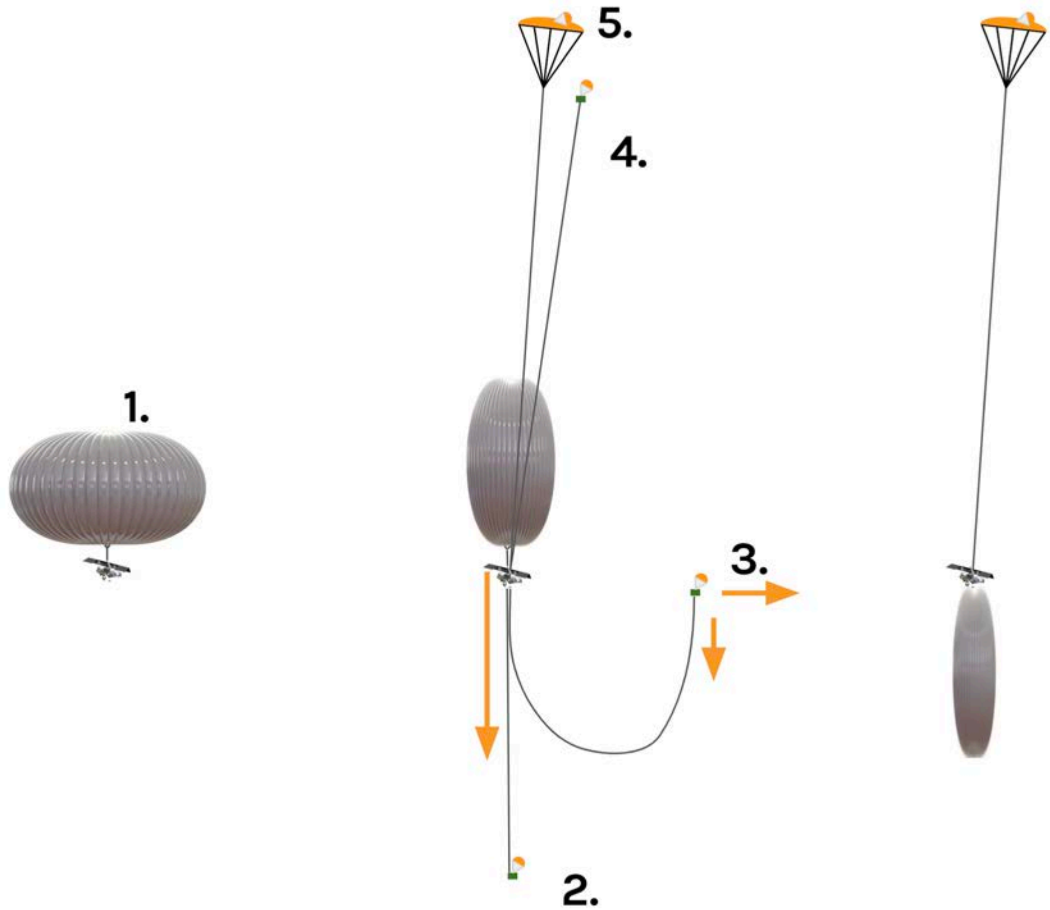


Figure 3-51 ***Stages of descent.***
Prior to termination and early in venting (left), parachute deployment (middle), descent under parachute (right).

The process went as follows:

1. The balloon was terminated, lift gas vented and the flight vehicle begins to descend.
2. The parachute container was lowered from the bus with the long tether connected to the bus spine.
3. A drogue attached to the container inflated and lifted the parachute container relative to the balloon while gliding out to the side away from the balloon.
4. Dynamic pressure on the drogue increased and stretched the tether above the balloon before the parachute container was opened.
5. The main parachute deployed.

Because the balloon would eventually vent enough helium that it would no longer be buoyant, it would invert relative to the bus and, in rare circumstances, could interact with the bus. This introduced risk to bus and payload components due to potential interaction with the balloon.

Bus-Separation Descent System

For the v1.6 vehicle this parachute design was replaced as part of the new bus-separation descent system. The goal of this system was to bring about a number of safety enhancements. A key focus was to remove the potential for interaction between the balloon and bus by disconnecting the bus during descent. This is a common practice in ballooning but Loon added a new element which triggered the disconnection at low altitude so as to avoid creating two descending vehicles in the airspace. Further, the separated balloon would incorporate all systems necessary to maintain safety in the airspace including satellite communication, GPS, flashers and transponder for tracking as well as pressure sensors for monitoring altitude and descent rate.

The bus-separation descent system brought about a number of safety-related enhancements including:

- Eliminating the inversion of the balloon during descent thereby mitigating risk to bus or payload components from the balloon.
- Individual parachutes for both the bus and the balloon.
- Increased parachute reliability due to static-line deployment of both parachutes.
- Configurable separation altitude which enabled some limited modification of the landing zone during descent.
- Post-landing disconnection of both the bus and balloon parachutes to reduce risk of dragging on the ground.

The bus-separation descent system worked as follows:

- Prior to termination, the landing zone ground level elevation was determined and the desired bus-separation altitude's corresponding atmospheric pressure setting was updated, overriding the default failsafe settings.
- Bus-separation from the balloon was set to occur at low altitude (~3,000 m above ground level) in a HALO (High Altitude Low Opening) maneuver so that it occurred below commercial aircraft flight lanes in order to:
    - » Have only one object descending in the upper airspace for ease of coordination with Air Traffic Control
    - » Keep landing zones for the bus and balloon close together
- The balloon was vented by the FTS to initiate the descent.
- At the programmed pressure, the bus disconnected from the balloon via a squib-actuated disconnect mechanism.
- As the bus separated from the balloon, a tether maintained stability of the bus until the bus drogue was deployed. The drogue provided stabilization of the bus until main parachute deployment. See Figure 3-55.
- The bus main parachute deployed via a squib-actuated ring release mechanism after a set pressure offset from the bus-separation pressure. See Figure 3-56.
- The balloon main parachute container was lifted by a drogue and rose above the balloon before the main parachute was deployed.
- After landings were confirmed a command was sent to disconnect the parachute for both the bus and the balloon.

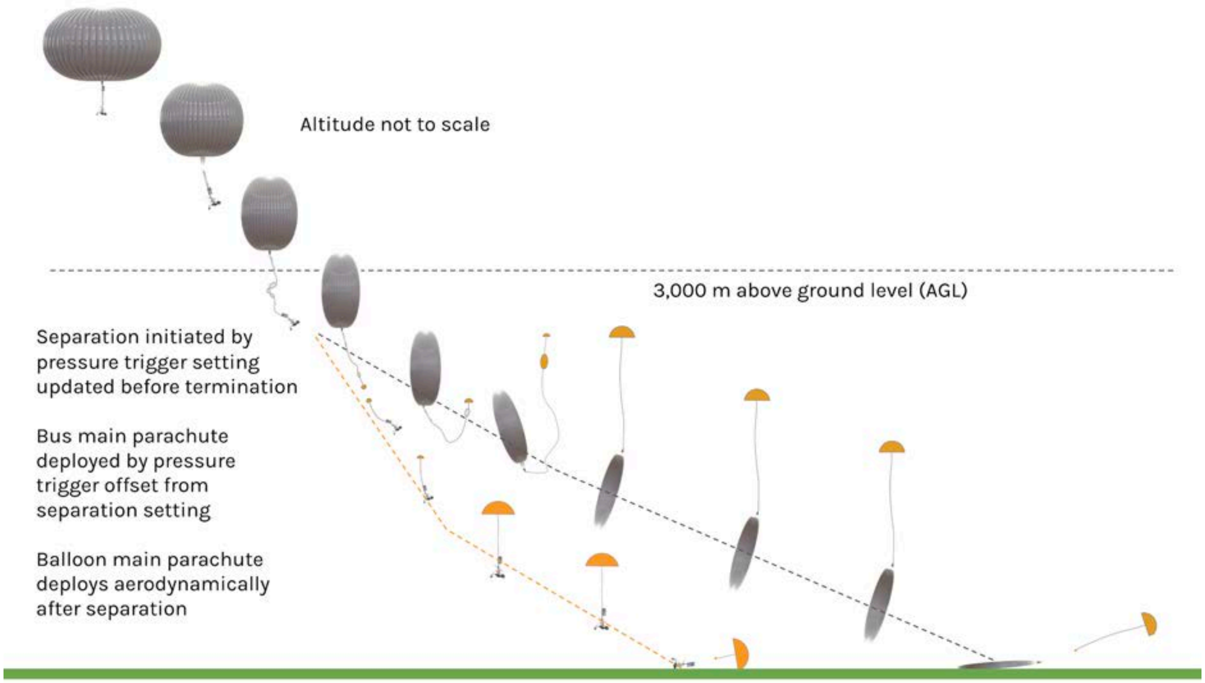


Figure 3-52 ***Nominal descent scenario 1.***
Bus separates from balloon at 3,000 m above ground level (altitude not to scale).

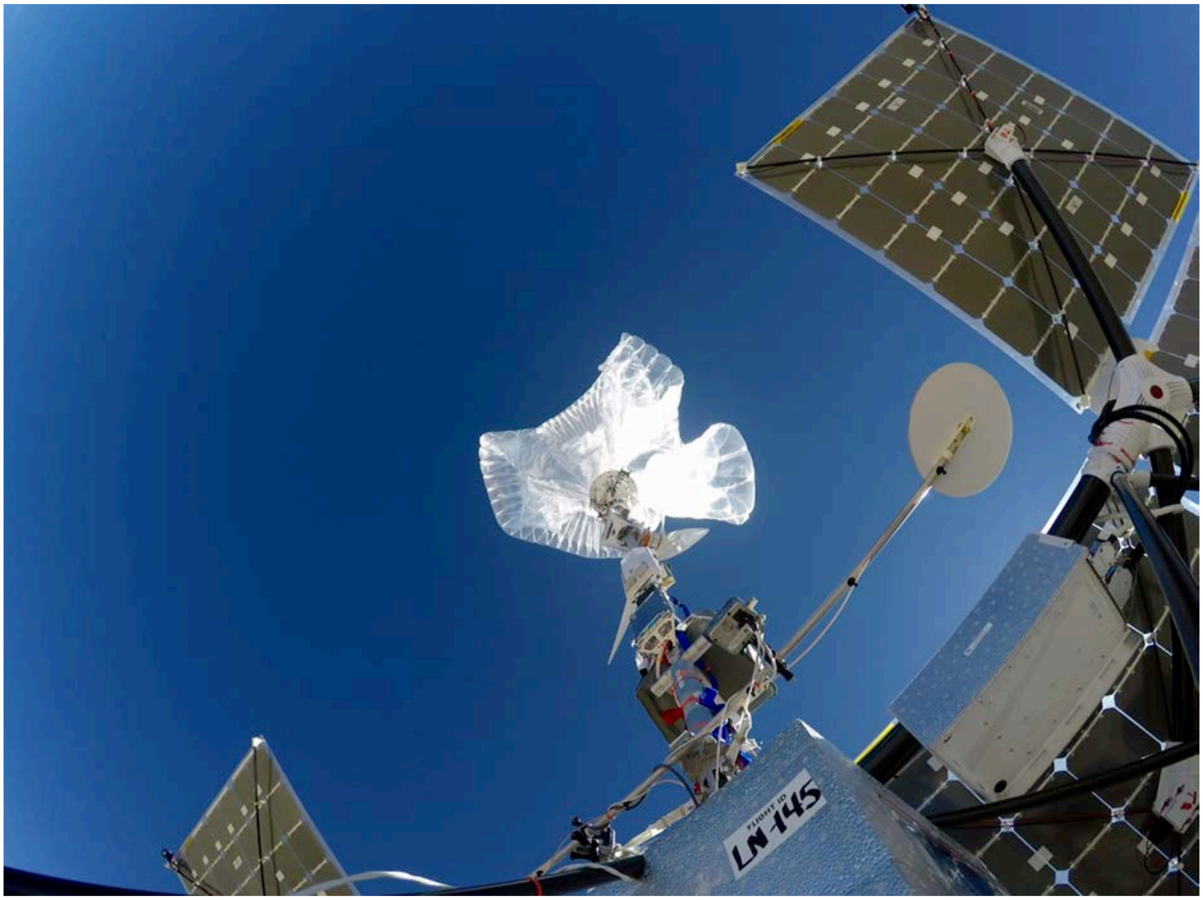


Figure 3-53 ***Nominal descent scenario 2.***
Prior to bus-separation, the balloon is mostly vented but still contains some lift gas and air ballast. This gas volume generates drag and keeps the balloon above the bus (avoiding balloon inversion).

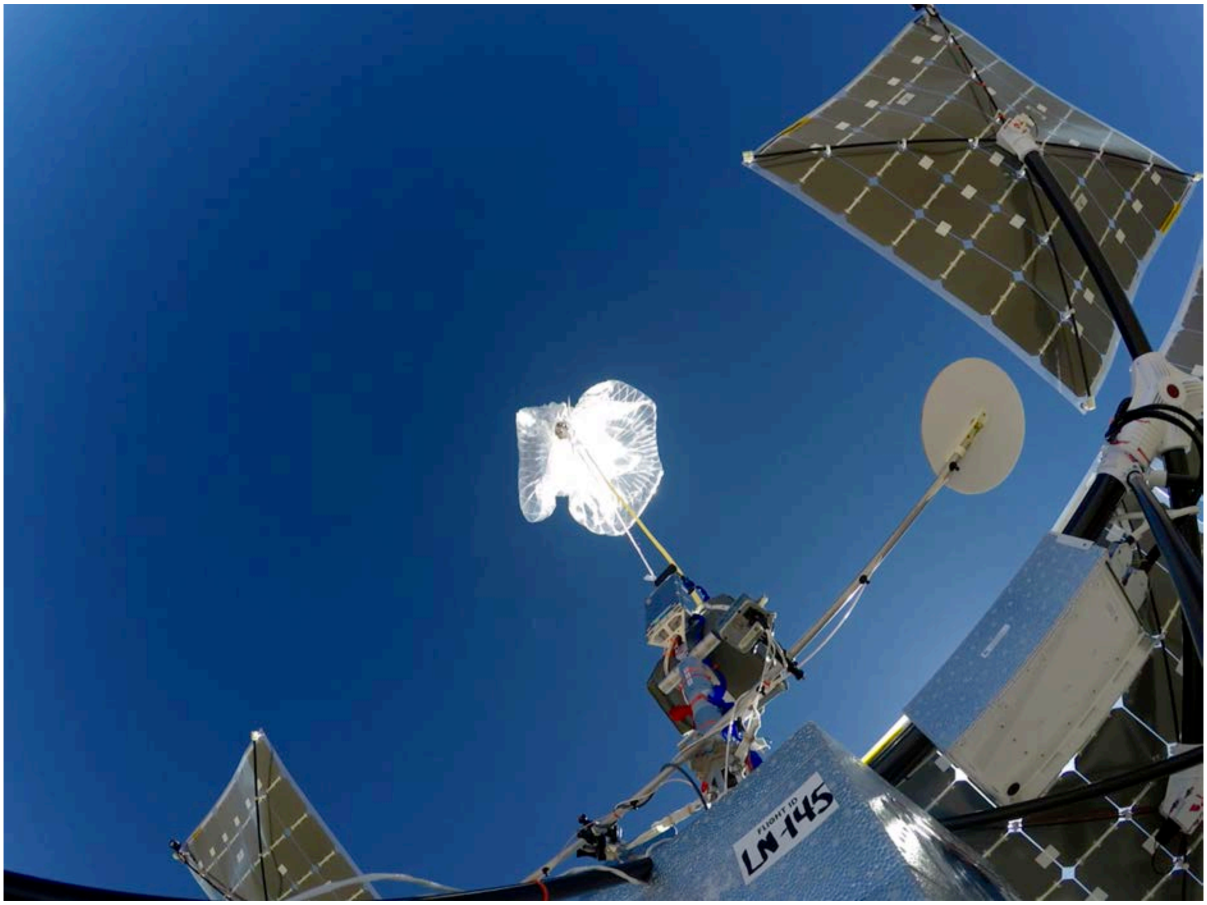


Figure 3-54 ***Nominal descent scenario 3.***
During bus-separation, the yellow static line and white stabilizing tether are visible.

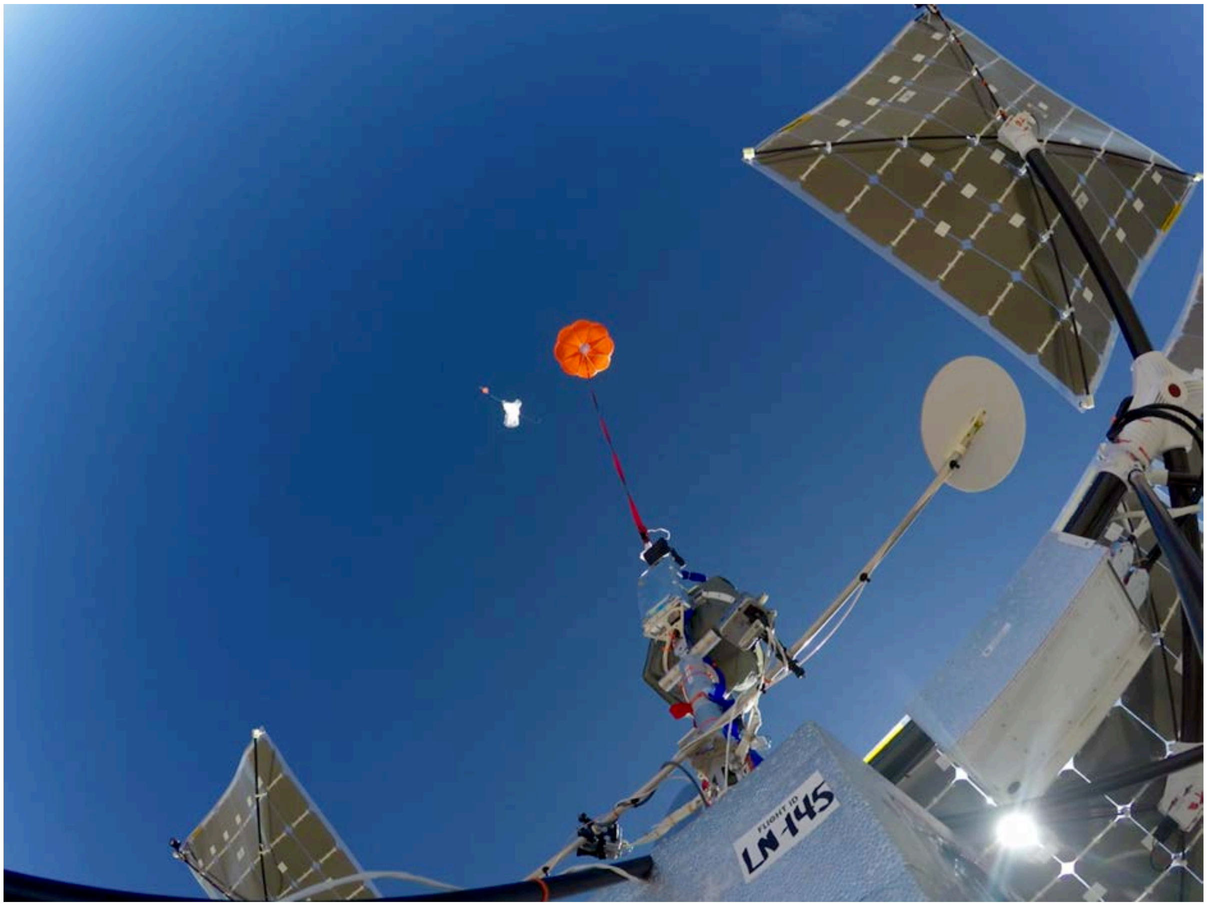


Figure 3-55 ***Nominal descent scenario 4.***
After separation, the bus drogue and balloon drogue are seen prior to main parachute deployments.

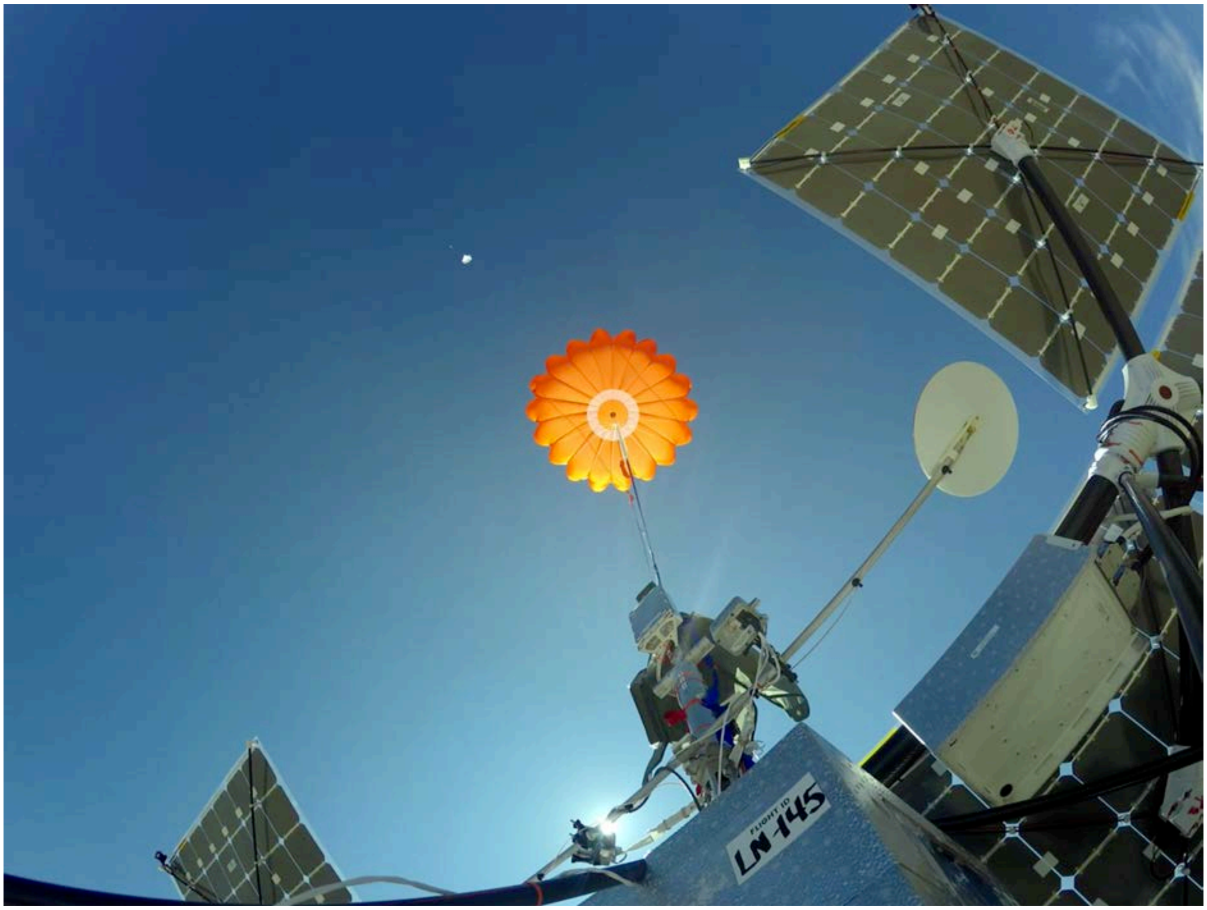


Figure 3-56 ***Nominal descent scenario 5.***
Bus main parachute and balloon main parachute deployed.

Descent Scenarios and Triggering

There are three key scenarios for descents which utilize failsafe triggering so that the bus-separation step and main parachute deployment step would occur at the proper altitudes and in the proper sequence. They are the planned landing, unexpected termination and rapid depressurization scenarios.

Just prior to a planned landing, the flight engineers updated the triggers to be:

- Bus-separation triggered at a rate of air pressure change of 120 Pa/sec (which would not be achieved during a normal descent) or atmospheric pressure corresponding to 3,000 m above ground level near the landing zone.
- Bus main parachute deployment triggered at the bus-separation pressure set point plus a positive pressure offset.

At float, the triggers were the system failsafe settings of:

- Separation would be triggered upon reaching 40 pa/sec rate of air pressure change or the pressure corresponding to 10,600 m altitude (whichever state occurred first)
- Bus main parachute deployment would be triggered at the pressure corresponding to 10,000 m altitude

If the flight vehicle were to suffer an unexpected termination with a command-and-control outage, the pressure triggers would activate at their corresponding altitudes and the sequence of separation, delay then the main parachute deployment would be maintained.

If the balloon were to rapidly depressurize or burst, the rate of air pressure change trigger would initiate bus separation within the first few minutes, before the risk of balloon instability increased, and the pressure trigger would initiate bus main parachute deployment at 10,000 m. The balloon main parachute would deploy in the minutes following bus separation. See the discussion of balloon stability below for context on why high altitude separation reduced flight vehicle risk for these extremely rare events.

In certain scenarios the separation mechanism could be deactivated to maintain a single flight vehicle all the way to the ground if desired. This could be paired with Super Soft Terminate so that landing speeds were low in this procedure. An example use case of this procedure is if a landing was forced to be on the water when the buoyancy of the balloon enabled complete flight vehicle recovery.

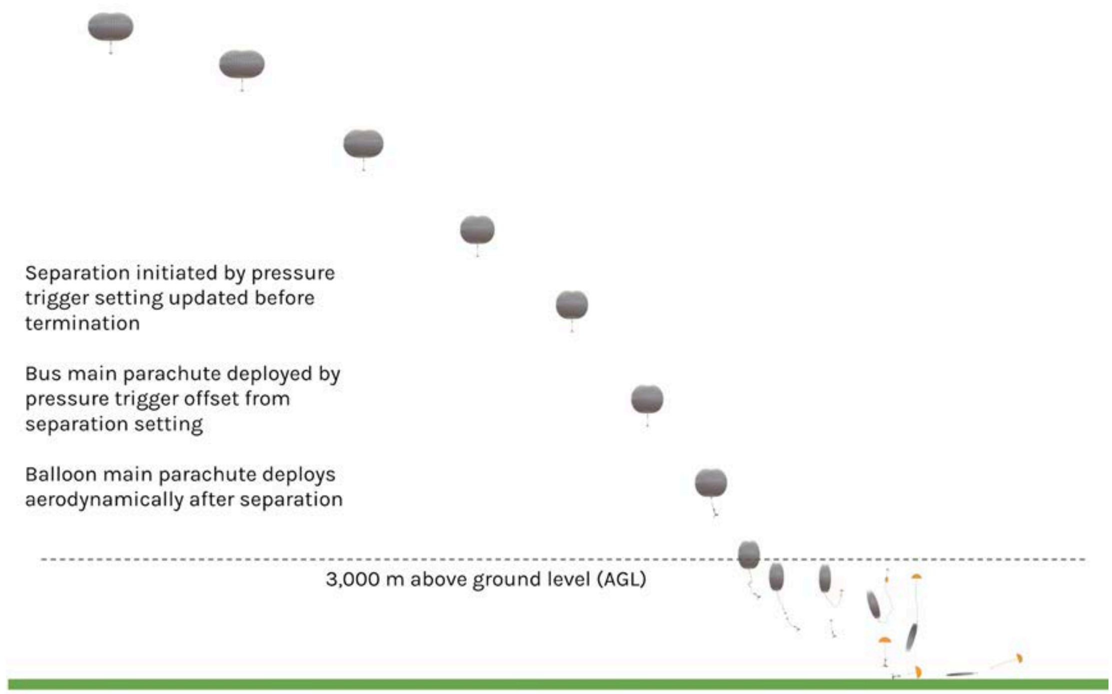


Figure 3-57 Planned landing scenario (altitude approximately to scale).

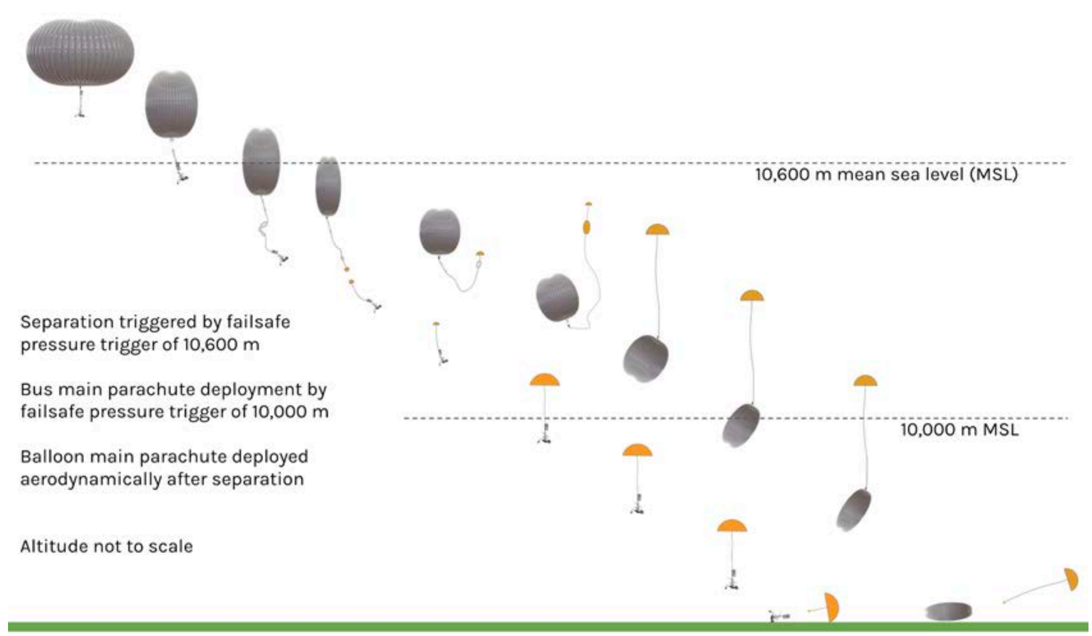


Figure 3-58 Unplanned termination scenario (altitude not to scale).

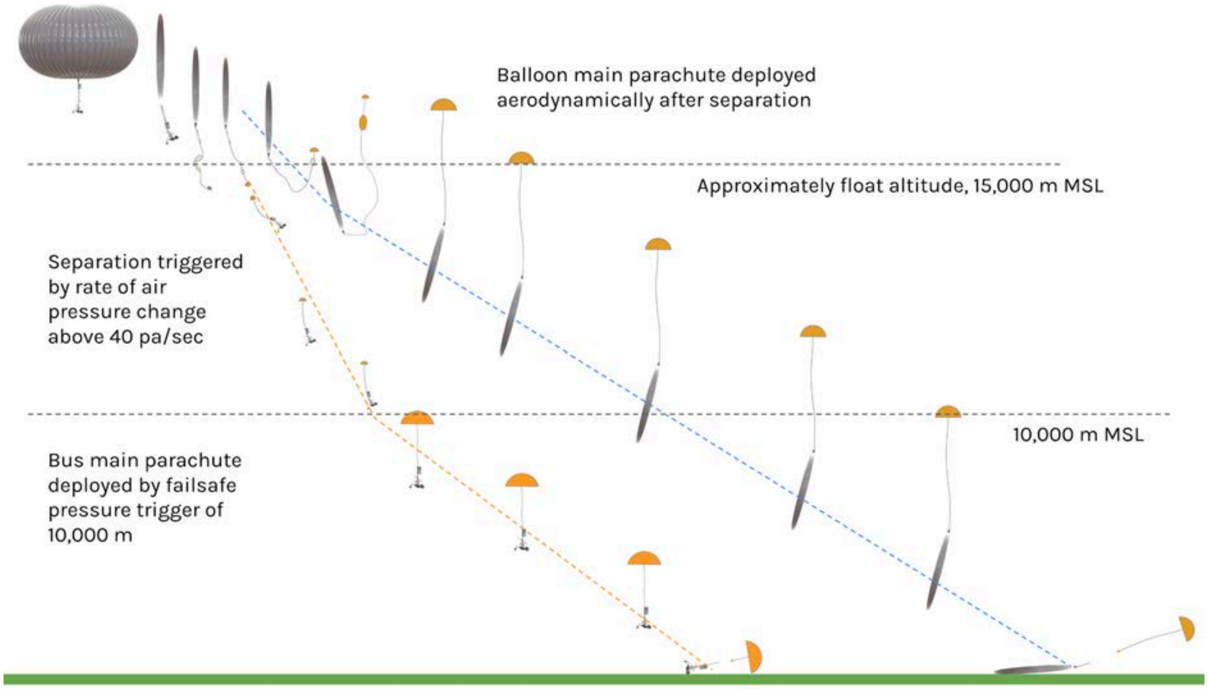


Figure 3-59 *Unplanned, rapid depressurization scenario (extremely rare).*

Summary of Key Learnings and Innovations

- Having multiple, different methods of termination with different venting rates enables a range of scenarios (with a very low speed landing termination method as a useful safety contingency)
- Bus-separation was essential for isolating unpredictable balloon behavior and interaction during descent
- Configurable triggering for different stages of descent (bus-separation, main parachute deployment) enables flexibility for operational evolution and special case contingencies

Landing Zone Prediction Accuracy

Accurately predicting the landing zone (LZ) was a key aspect of operational safety. Prior to termination of the balloon the fleet management system would generate predicted descent trajectories for any point on the predicted flight path that the flight engineers chose. This allowed for pre-planning of a descent time still hours in the future.

The simulated trajectories would be generated by merging two models to generate a simulated descent trajectory:

- A model of the vehicle dynamics as altitude vs. time
- A model of the wind column's influence on the flight vehicle as wind speed and direction vs. altitude

Once merged this generated a simulated trajectory which would predict the zones on the ground where the bus and balloon would land. The resulting LZ prediction accuracy was tracked as a circular error probable (CEP) 99% distribution. This was a key metric in the operational safety case controlling landing errors to a certain accuracy but limiting outliers to a 1% threshold value.

One method of increasing LZ prediction accuracy was to simply descend more quickly. This reduced the time that errors in the wind speed and direction vs. altitude model would accumulate. This was achieved via different methods of rapidly venting the balloon using hot gas generation to melt an opening larger than the feature plate openings for the rotary cutter FTSs. The resulting LZ prediction accuracies were many times better than previous performance for CEP99%.

This method created a large opening in the envelope gores during descent, which could further enlarge through the flapping of the balloon continuing to tear open the gores. Thus, this method decreased accuracy of vehicle vertical profile modeling and so relied primarily on reduced descent duration to improve LZ prediction accuracy (which it did very successfully).

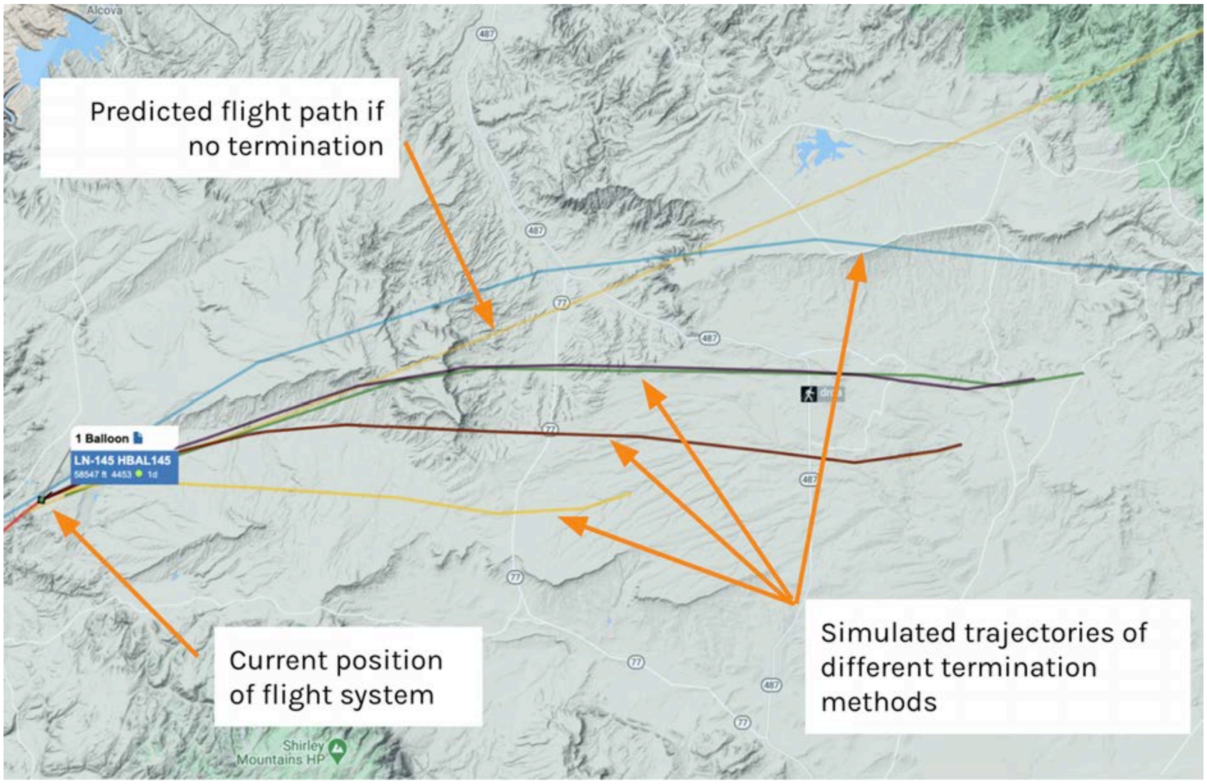


Figure 3-60 ***Pre-termination simulated descent trajectories.***

A key challenge that resulted from using this method was the need to stabilize the balloon once it had fully vented all lift gas and air ballast. While there was a reasonable amount of either gas remaining in the balloon it would retain enough drag that it would form a vertical train of tension from the bus up to the balloon. Once the balloon vented the gas below this level it could lead to a condition where the terminal velocity of the balloon compared to the bus could change back and forth resulting in the risk of the balloon entangling the bus.

Figure 3-61 ***Special test flight showing the fully vented balloon in vertical train above the bus at upper altitudes.***

Figure 3-62 Special test flight showing momentary increase in balloon terminal velocity relative to the bus at lower altitude.

To prevent this, a balloon stabilizer was explored using either a drag disk or drogues deployed from the top apex assembly. Both systems were challenging to make sufficiently reliable due to entanglement with the balloon as it fully vented and was able to interact with the top plate.

Figure 3-63 Drag disk prototype ready for a test flight.

Without a reliable balloon stabilizer to enable rapidly venting the balloon safely and producing a short descent duration the limits of LZ prediction accuracy needed to be quantified. In particular the scale of the effect of potential wind prediction errors needed to be assessed since improving it would take a sounding system such as a dropsonde.

In order to isolate and quantify the effect of wind model error, the simulations could be re-run for past descents using the flight's actual vertical profile data in order to produce the irreducible wind model prediction error.

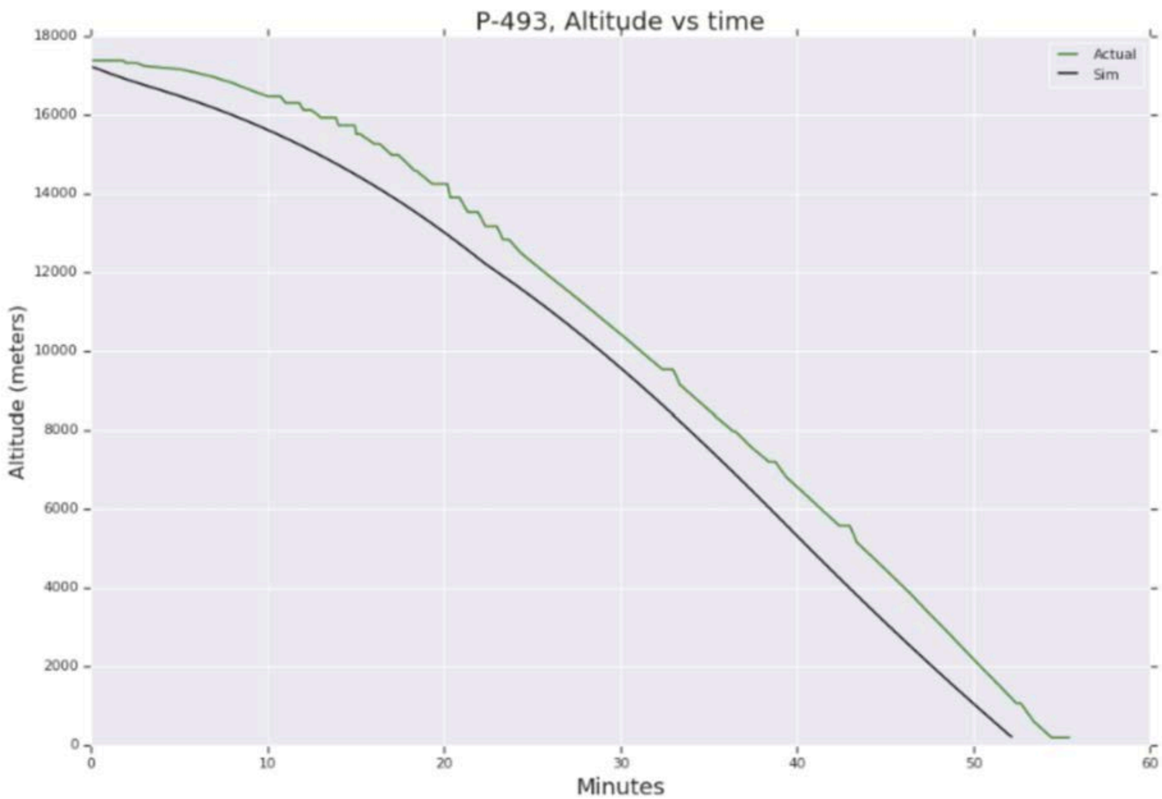


Figure 3-64 Example of vertical profile (altitude vs. time) prediction and actuals.

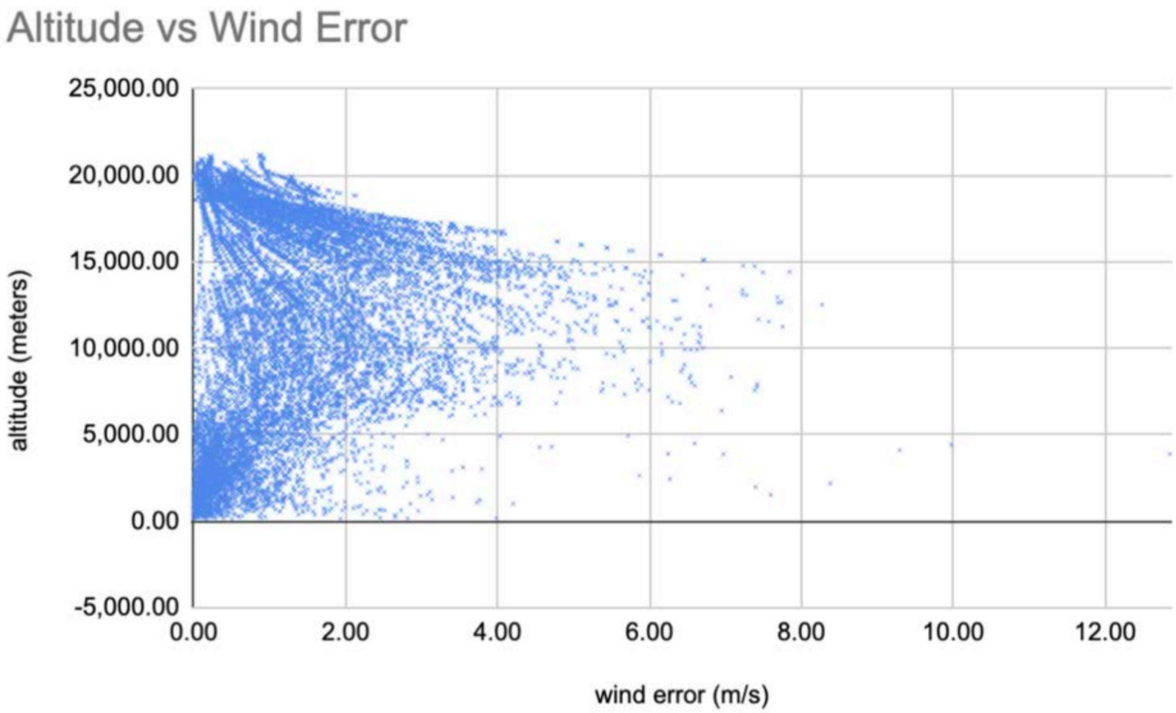


Figure 3-65 ***Example of wind speed error versus altitude data.***
(Direction not represented in this plot). Note that prediction is best at ground level and at the starting location of the balloon descent. Without the balloon's own data, the error would continue to grow as the altitude increases past 10 km due to the increasing scarcity of observations at those higher altitudes.

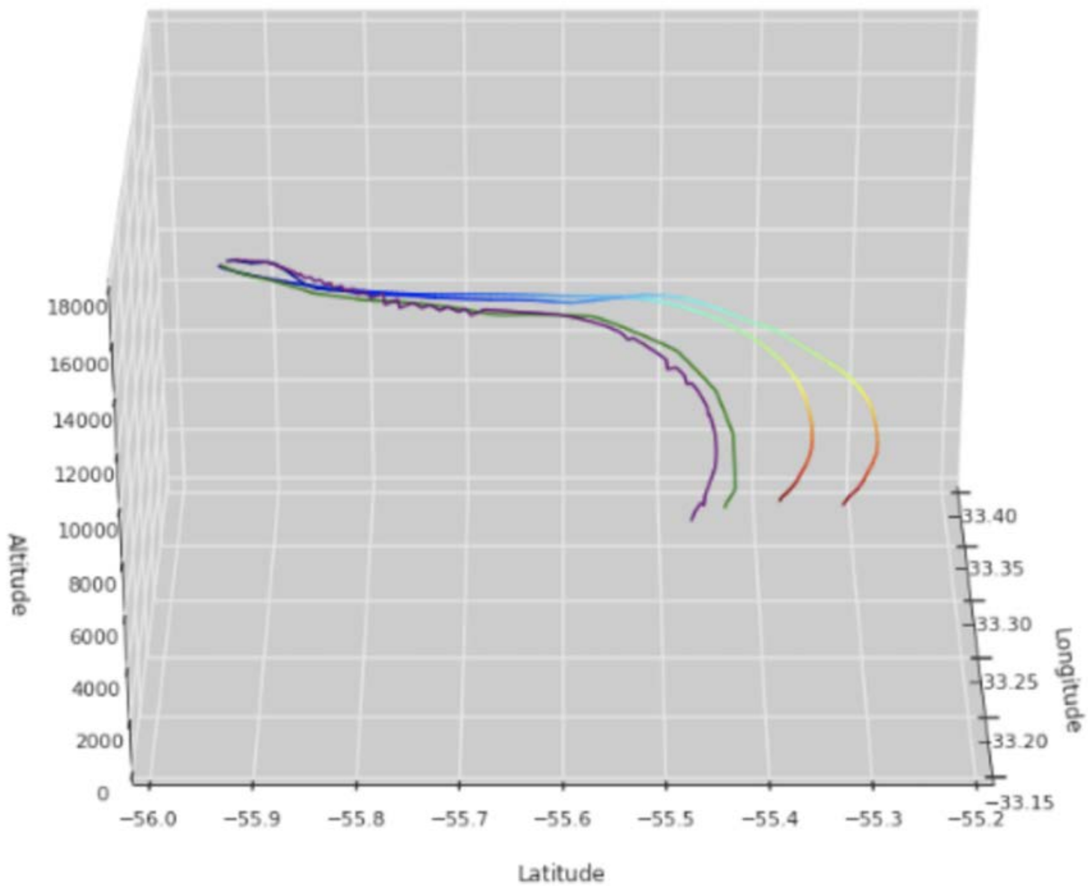


Figure 3-66 ***Resulting irreducible trajectory error due to wind model error.***

The scale of wind speeds and variability in direction varies by geographic region and time of year. In regions or simulations with relatively little change in direction and high wind speeds the prediction errors could be higher than tolerable. In regions or simulations with significant change in direction and lower speeds the errors were generally far lower as they tended to cancel out. This effect could be utilized by quantifying the cumulative change in direction and windspeed throughout the simulation and to create a risk metric for outlier accuracy behavior for a given trajectory.

An emerging approach at the time Loon wound down was improving prediction confidence using a Monte Carlo method to develop a probabilistic landing error map given the shape and scale characteristics of the simulated trajectory prior to termination.

Some key lessons from this analysis include:

- Predictability in descent rate profile is essential for accurate landing zone prediction with modest descent rates. But, in some cases, wind prediction errors may still dominate.
- For landing zone prediction, faster descents are better because accumulated error scales roughly with descent duration.
- One critical limiting factor in speed of descent is maintaining balloon stability in order to prevent collision between the balloon and bus. In the rare case of rapid depressurization, separating the bus from the balloon at high altitude can remove this internal collision risk but results in an additional descending vehicle, increasing risk and burden on the air traffic control system.

FTS Squib Triggering Logic

Loon put a significant amount of effort into the squib triggering logic both in hardware and in software for termination/cutdown and later bus-separation. For cutdown, we always erred on the side of cutting down to prevent an uncontrollable balloon. For bus separation, we ensured the vehicle could safely descend and separate without any input from humans or systems on the ground.

For firing hardware, we charged a bank of supercapacitors to ensure that we could provide enough current to fire the squibs, even if input power had been removed from the board. In cases like bus-separation, we also added a backup battery local to the board so that the board could operate for multiple hours without input power.

At the heart of the logic, we had two important concepts: arming and triggering. Arming was the process of charging the supercaps such that they could be triggered. Triggering was done by a number of configurable logic blocks based on which squib was being fired. For all squibs, a *By Command* trigger was used, such that a human on the ground could send a command and ask the squib to fire, but often triggers were added for things like command and control outages, rate of air pressure change (indicating vertical velocity) for parachutes, and other reasons.

This figure shows roughly what the logic would look like for balloon termination/cutdown.

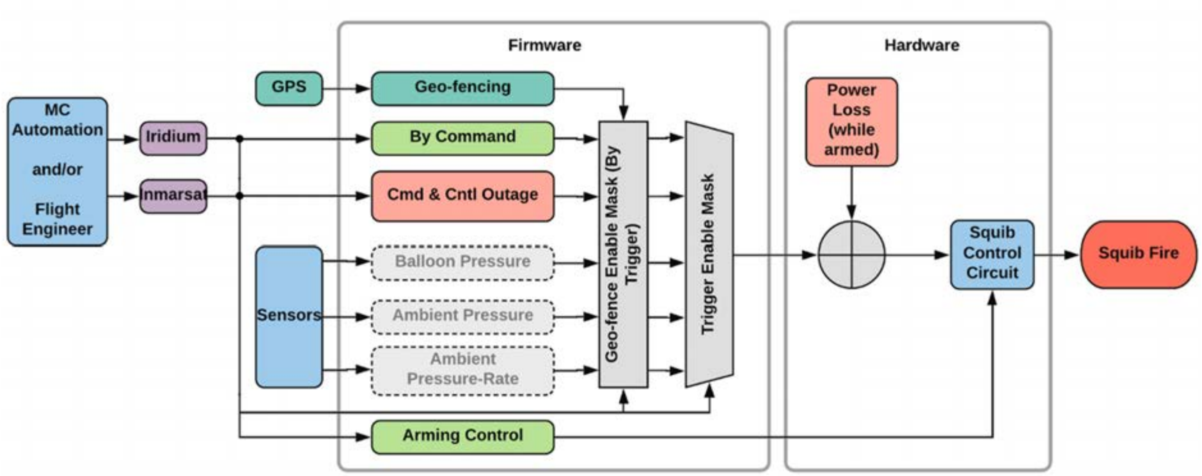


Figure 3-67 Squib firing logic.

For flight termination & parachutes on the v1.4 systems, Loon used the following logic:

- The Soft and Super-Soft channels were armed before launch with By Command and Command & Control Outage triggers. The Command & Control Outage trigger allowed us to specify a maximum time the balloon could stay afloat without successfully sending a packet over either the Iridium or Inmarsat satellite networks. If we were unable to send a packet (the packet did not have to make it to our data center- only be accepted by their satellite), the balloon would self-terminate. This allowed us to ensure that in the case of balloon-side hardware failures of both systems, we would not remain in the air uncontrolled. Loon found the optimal time window for this trigger was three hours, as it allowed enough time for a network issue to recover, but not too much that the balloon would drift to an unfavorable location.
- The geo-fencing logic was activated such that if any of the active triggers were commanded over a predefined map of keep out zones (mostly populated or sensitive areas) the balloon would not fire the squibs.
- The Hard Terminate squib channel was armed and fired by a human during normal descents to obtain a faster descent rate.

- The Parachute squib was also armed on the ground, with By Command and Rate of Air Pressure Change triggers, such that the parachute would automatically deploy on descent when the speed of the system descending through the air was sufficient for parachute deployment.
- Any armed squib channel would also fire in the case of power loss to the board that controlled it, after about a 1 minute delay to ensure the loss wasn't transient.

For the v1.6 vehicle that introduced bus-separation, this logic became much more complex, as Loon also had to use altitude (pressure) triggers, as well as introducing a dynamic pressure trigger (rather than simply rate of air pressure change), but overall, the concept of arming, triggers, and firing remained the same.

General Design Approach

The high-level design goals for all of Loon's subsystems were:

- Meet environmental and architectural constraints
- Maximize serviceable lifetime of the flight vehicle
- Minimize power consumption and mass
- Maximize revenue gathered (e.g., GBs delivered) per service hour
- Minimize cost of vehicle and ground stations
- Anticipate future generations
    - Modularity
    - Expandability
- Maintain flight vehicle safety throughout use and evolution

A constant question and tradeoff for Loon was the build vs. buy decision for certain components. Among the many considerations into this decision were a number of key factors to consider:

- Stratospheric environmental (temperature, pressure, UV, such conditions were extreme compared to many OTS component specs)
- Mass and power (since both were precious resources for the flight vehicle)
- Performance

- Cost (very important factor for Loon)
- Development time and reliability or validation testing

Early on, when Loon was iterating the design of the flight vehicle quite often while refining the commercial vehicle, it was less crucial to have optimized components so speed and ease of development were prioritized. As Loon began to stabilize the commercial flight vehicle, designing more optimized solutions made sense.

Of all factors, environmental conditions tended to be the leading reason for Loon to build components, followed closely by speed and cost of development. However, later in the project as Loon began scaling up the commercial fleet, flight vehicle cost was an increasing focus area.

Mechanical

Surviving a Burst Loading

If the balloon ruptures due to an extreme event, the balloon bottom assembly will accelerate sideways at hundreds of Gs due to the stored energy of the superpressure in the form of tendon tension. The bus, below the downconnection, must be isolated from the acceleration of the bottom assembly. If it was not, the bus and most of its hosted assemblies would need to be much more robust, making them significantly heavier, which is a huge penalty to the overall system design.

Loon's downconnection incorporates a mechanical fuse to limit the bus to <10G accelerations. This separates the downconnection near where it attaches to the balloon bottom plate and through the use of energy absorbing rip-stitch tethers, keeps the system together while isolating loads.

Loon refined this design several times to optimize for predictability and to support a greater bus & payload total mass. The latest design is shown below, which uses a moment fuse bolt as a precision part design to yield at a specific moment but with pins accepting the normal torsional loading from despin.

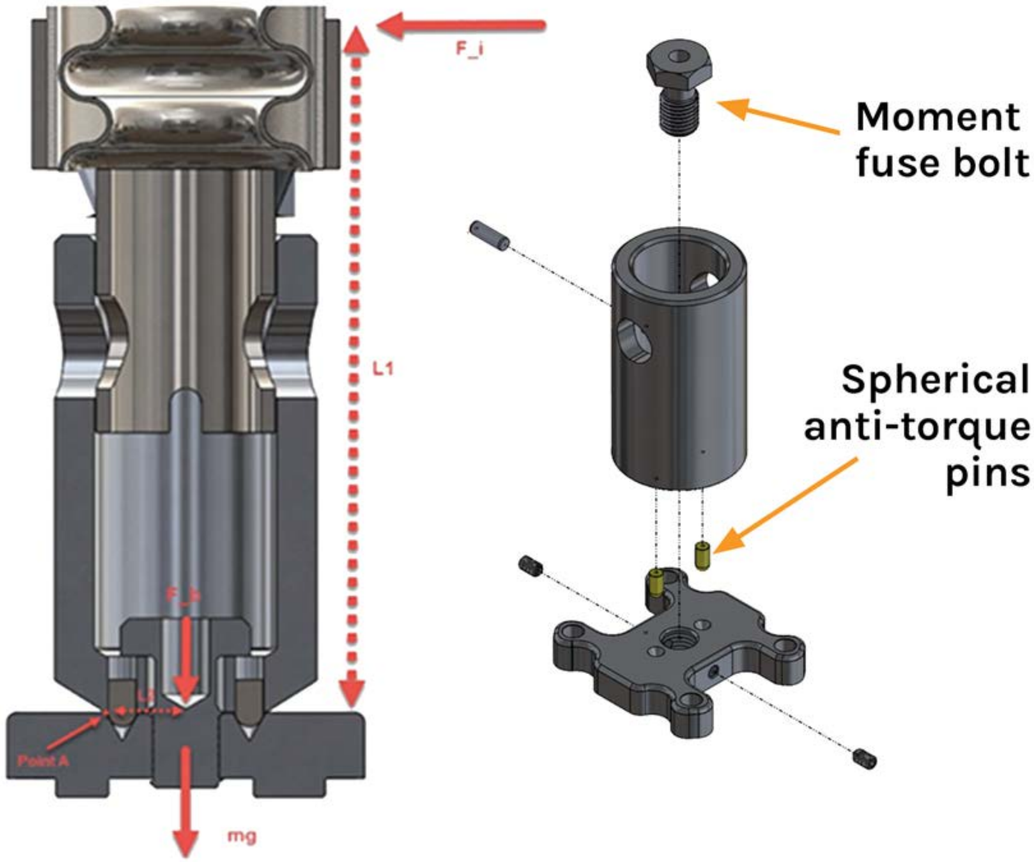


Figure 3-68 Mechanical fuse assembly.

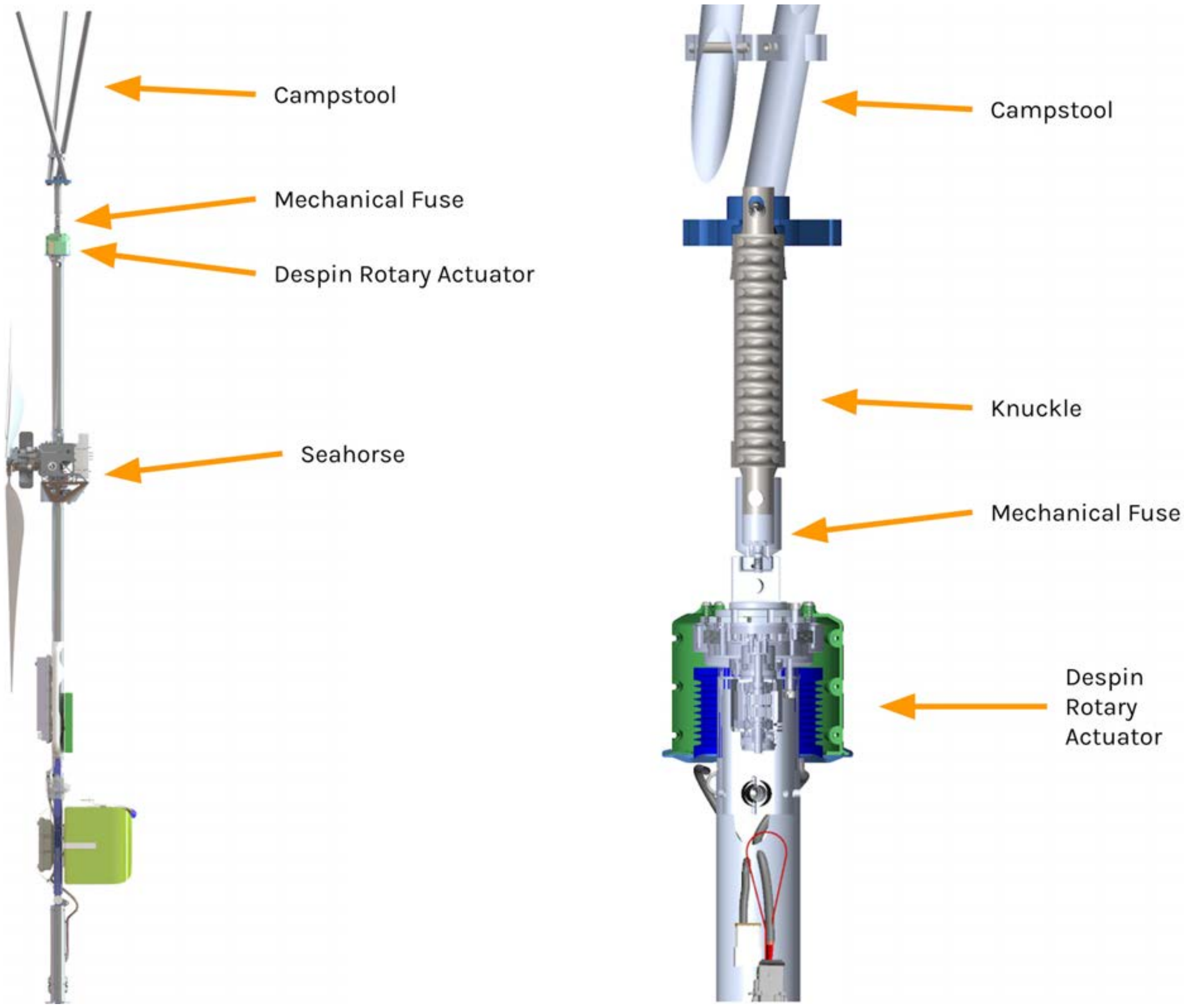


Figure 3-69 Mechanical fuse placement at top of the downconnection below the campstool connection to the balloon bottom assembly.

To test and validate that the system works, it is tested on the ground using an air cannon capable of producing hundreds of Gs of acceleration. This air cannon, nick-named Beowulf, can also be used for testing other components to insure they survive burst loadings.

See the related Artifact montage video of Beowulf firings (LnArt- Beowulf Testing Video.mp4)..

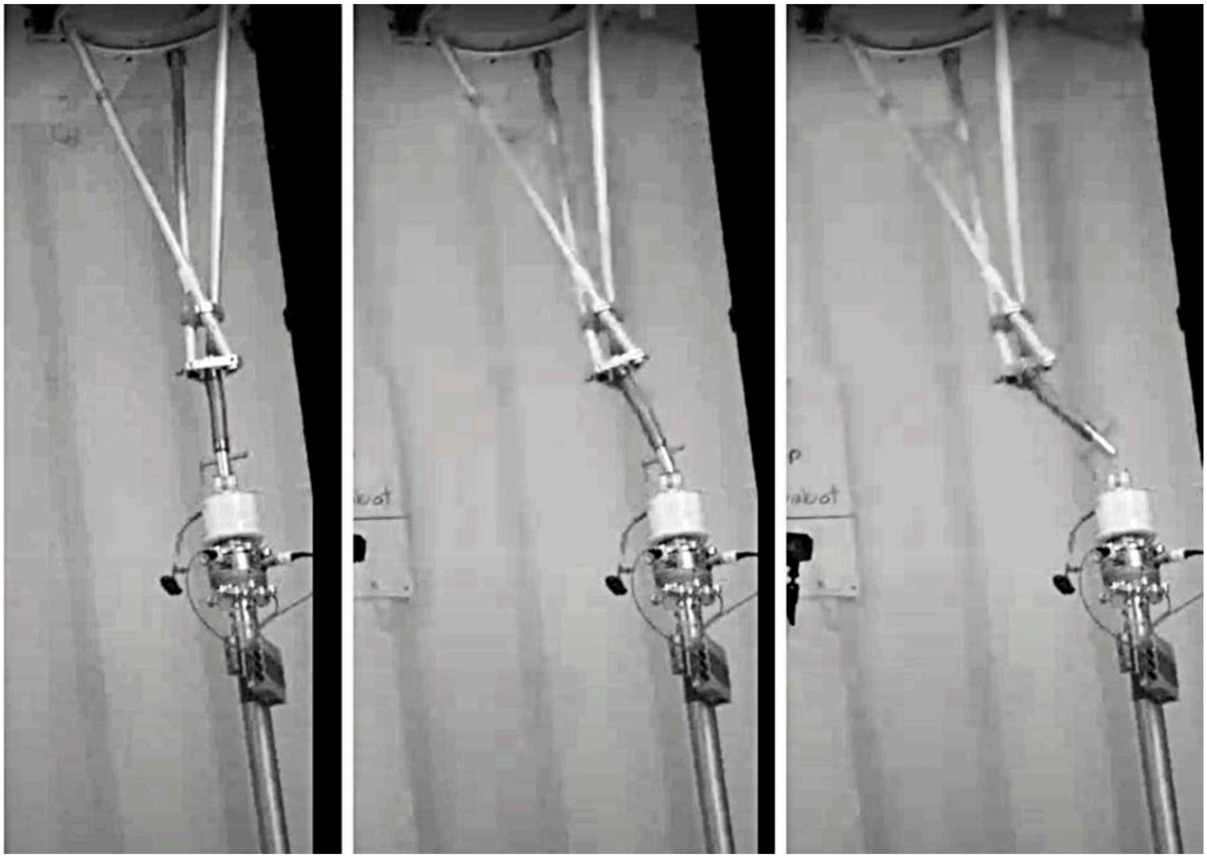


Figure 3-70 High speed camera images of ground testing the mechanical fuse at 300 G.

As a final mitigation many components (especially components that have not been fully tested) are also tethered to ensure they stay attached to the vehicle at all times.

Bearings and Lubrication

The lower limit of operating temperatures meant that most greases would be working significantly below their pour point (typically -50°C). The low viscosity was usually unacceptable for starting and often operating conditions without heaters, while pure oils lacked film longevity and adhesion. It required using several different solutions chosen specifically for each environment and design case. For example, for low RPM range solutions, Loon often chose dry lubrication, with graphite impregnated polymers or tungsten disulfide on non-oxidizing stainless steels. On medium RPM devices, chopped PFPE based greases were very successful, Loon used 8951R, where the dynamic viscosity over the RPM range was amenable to the use case, and adhesion was acceptable. For high RPM devices such as the ACS turbomachine, which required a good lubrication layer at high temperature, Loon leveraged specialty ester based greases, which sacrificed ideal low temperature film behavior for optimal performance for long term reliability for bearings. This specific use case was compensated with special firmware for start conditions.

Thermal

The stratosphere is a unique thermal environment in that it is both very cold, as well as low ambient pressure, which makes it challenging to dissipate heat produced on the vehicle through conduction as you would on the ground. This fact made it extra important to design adequate heating and cooling solutions for systems on the balloon, as well as accurate sensing to better understand the thermal condition to control the system and refine the design.

Sensing

Loon had a lot of thermal sensors, for measuring environmental conditions, internal gas temperatures and temperatures of components in the system.

For measurements of the ambient air, as well as the temperature of the helium in the balloon, Loon used a very small 4-wire RTD with some heat shrink around it that was metalized with aluminum to reject solar radiation during the day. It was especially accurate at night (with less than a degree of error), and reasonably accurate during the day as even with the metallization it was skewed by radiation from the sun. Loon fused the sensor with NOAA data to determine the best estimate of temperature to use in the balloon models.

In general, Loon tried to standardize on RTDs over thermistors for sensing of other hardware on the balloon, like motors and mechanisms. RTDs are more expensive but provide a more uniform accuracy over the temperature range (especially on the cold side where thermistors become super high resistance).

For the electronics, each board monitored its temperature closely using a I^2C-based temperature sensor in various locations. Temperature monitoring allowed Loon to run a control loop to heat the boards to keep them above their rated operating temperatures using integrated onboard heating resistors thermally coupled to the ground planes.

Photon Harvesting

While it may seem counterintuitive, cooling of systems can also be challenging in the stratosphere due to the low air density making convection much less effective than at lower altitudes. Cooling was particularly a problem for devices like solar maximum power point trackers that produce heat during the day, but not at night, yet need to be kept warm to avoid damage. A novel solution used was something Loon called photon harvesters. These were essentially white plates (with high emissivity) that pointed towards the earth. During the day, they were able to reject heat to the surrounding air (through conduction and convection), but they also harvested upwelling IR radiation from the earth (which is infrared radiation produced due to the inherent warmth of the earth), which provided a small heat source to help contribute to the heating power required at night and reduce the energy spent to keep the boards warm.

Related artifacts include:

- *LnArt- Photon Harvesting*
- *LnArt- Survival Heater Power Derivation*
- *LnArt- Thermal zones on Project Loon*

Heating and Preheating Systems

Since finding grease that works (and allows for high reliability systems) in very cold temperatures is challenging, on newer systems Loon started implementing a preheat of moving parts to help ensure they did not attempt to operate at extremely low temperatures. One example of this issue was on the motor controller for the lateral propulsion system, which Loon designed to run current through the phases of the motor without spinning the motor to always warm the motor up to -60°C before spinning it up.

In other cases, where off the shelf parts, such as IMUs, were required, Loon applied external heaters to the devices and regulated them to a stable, safe temperature to keep them within their target operation range.

Electrical

Loon's electrical system was largely custom designed for specific use cases, as most off-the-shelf hardware will not function reliably in the stratosphere without significant repackaging. In general, Loon had a philosophy to use mostly industrial rated components that were rated down to -40°C (rather than mil-spec), then heat the boards and assemblies to keep them within their intended operating range. In addition to temperature, hardening the electronics to survive storm induced transients also proved critical.

The evolution of Loon's system can be seen in the artifact LnArt- Loon Schematic.pdf.

Hardening Systems to Storm-Induced Electrical Activity

The two important symptoms of dangerous weather are *rapidly changing electric fields* caused by nearby (but below the balloon) cloud-to-cloud lightning and *turbulent winds*, both usually associated with convective activity / thunderstorms.

Loon found out fairly early in the balloon development that the dangerous storm weather factors would be an issue, and that we would need to make the system robust to this activity. After some research and talking with experts in the field, we determined that the electrical activity of concern to Loon is not direct lightning strikes to the balloon (which, while not impossible, were less likely at our operational altitudes), but rather large static electric fields as well as large, rapid changes in ambient electric fields during lightning discharge. Clouds develop a large static charge buildup due to charge separation; this static field can cause corona discharge and small lightning streamers on balloon components. When a lightning flash occurs, a large amount of charge flows to ground or neighboring clouds, creating large and rapid changes in the electric field in the vicinity of the balloon. The high dE/dt induces large displacement currents on the balloon wiring and structure. Specifically, the CAN and power cable that ran from the bottom to the top of the balloon (~20 meters) acted like an antenna and captured large electric field changes and generated streamers from the top plate and payload. This implied that the top plate of the balloon was most likely to see effects of storms, which correlated with what we observed.

StormTrooper

To better understand and quantify what effects Loon was seeing and to better design the protection elements, we created a board called StormTrooper. The board contained a number of onboard sensors designed to help quantify transients seen on the balloon so that we could adequately test for them on the ground to determine if the protection mechanisms were sufficient.

Two of these boards were included on each system — one on the bottom plate of the balloon near the ACS and the other on the payload truss. Sensors included are described below.

A ***peak current detector*** was implemented with a Rogolsky coil around the cable running from the bottom plate to the Apex assembly of the balloon, measuring the effective induced currents.

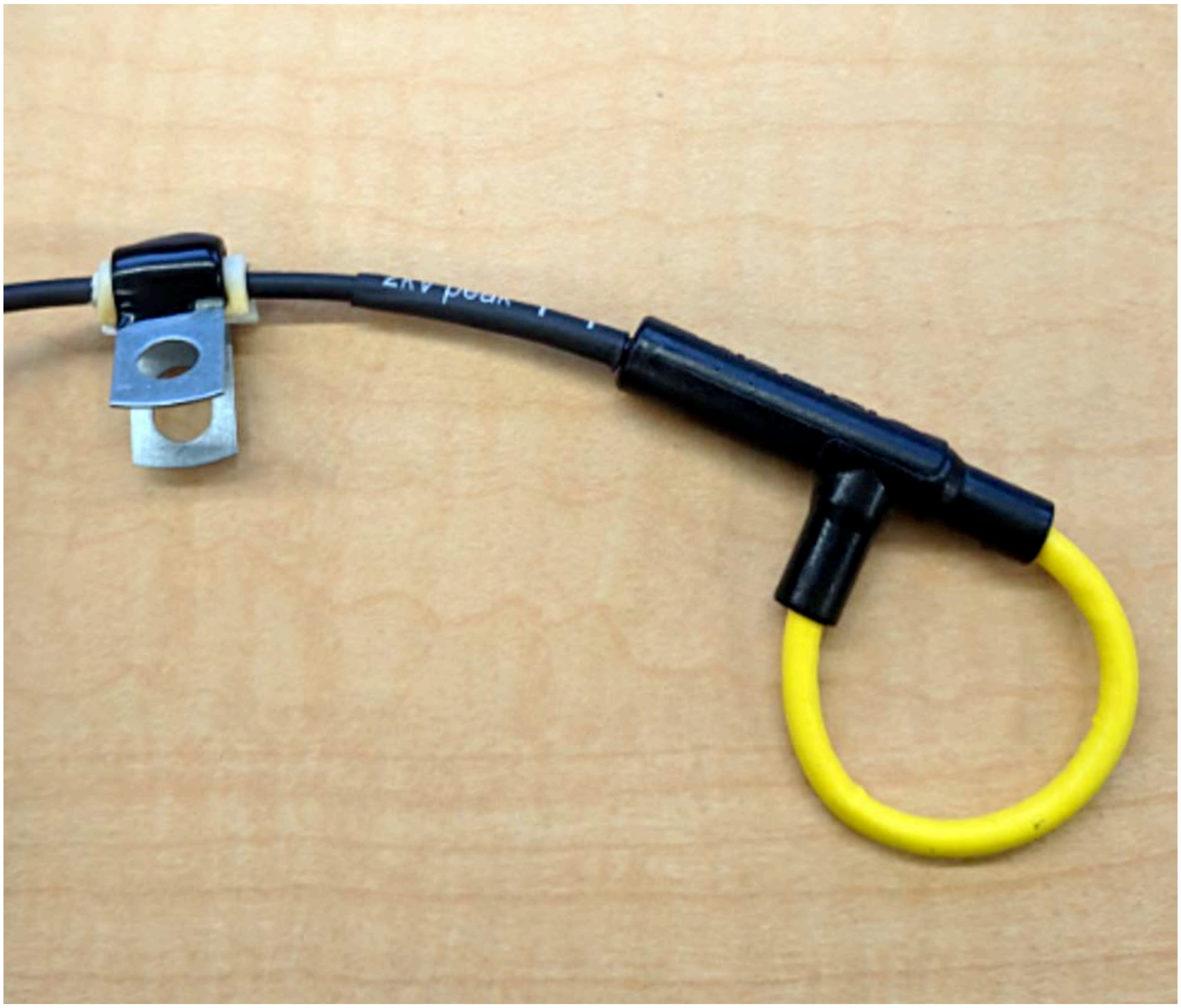


Figure 3-71 Peak current detector.

This sensor provided quite interesting results, as we were able to confirm current actually flowed on the cable during extreme events. We expected a wider distribution of smaller (< 1000A) transients, but actually found fewer than expected. Transients over 500A appeared to be quite rare, and when they did occur, were quite large (over 1.5kA), and typically resulted in a system failure on the top plate electronics.

A ***corona discharge current detector*** was implemented as a wire that hung beneath the truss (furthest object beneath the payload) with a small spherical antenna at the bottom, designed to measure induced corona currents (up to 10 uA).

Overall, Loon found this sensor to be more effective than expected in detecting storms and, in fact, experienced quite a large amount of discharge when anywhere near a storm. Unfortunately, we also found that there was a reasonably high frequency of streamers, which maxed out the sensing circuitry and prevented us from estimating electric field strength from the data. However, this sensor became quite an effective detection mechanism for nearby weather and could be used to prompt the fleet management system to proactively ascend to get away from the storm. There was a very strong correlation between this sensor and the ground-truth GOES-16 lightning-detection data sources. This was valuable for operation in the eastern hemisphere where GOES-16 does not provide coverage and it became one of the primary debugging tools for determining if a flight was being impacted by lightning.

A ***Moisture detector*** was implemented in both locations as a 0.5 x 0.5 inch patch of interlaced traces separated by about 6 mils, designed to increase in resistance if moisture condensed on the board.

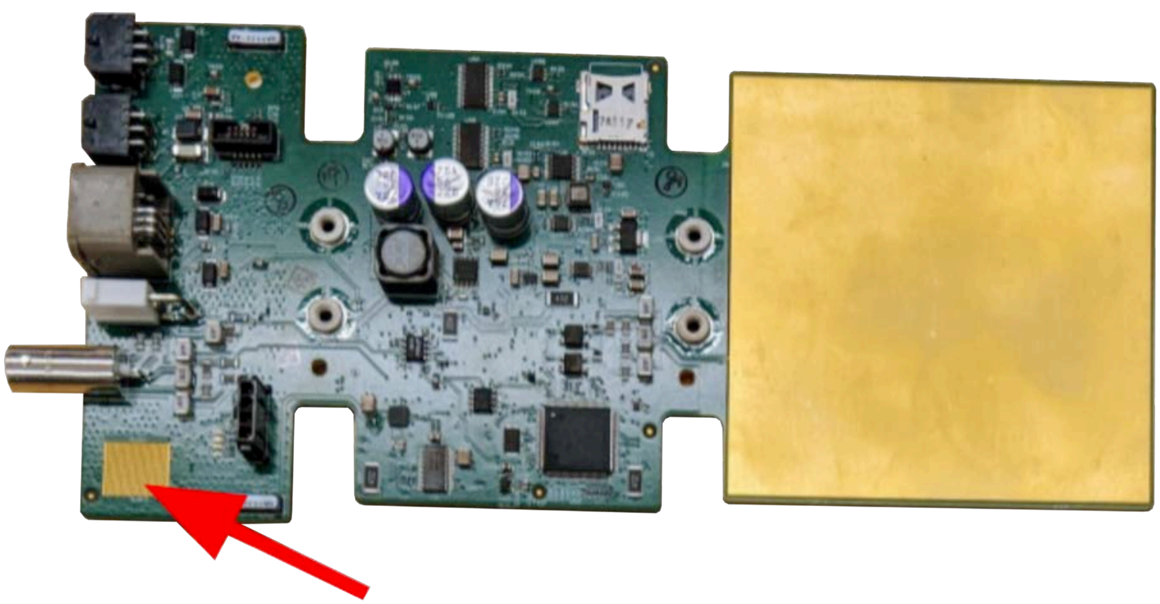


Figure 3-72 StormTrooper board moisture pad.

Loon added this sensor, which addressed the concern that there may be some water accumulation on boards from tall clouds that would cause intermittent shorts. We had experienced some issues with the CAN transceivers intermittently bringing down the network. To date, we have not detected any moisture with these pads, so it was likely not a root cause of any issues.

An off the shelf ***lightning detecting IC and antenna*** was included, originally designed to detect thunderstorms within a 40 km range, for applications such as golfing.

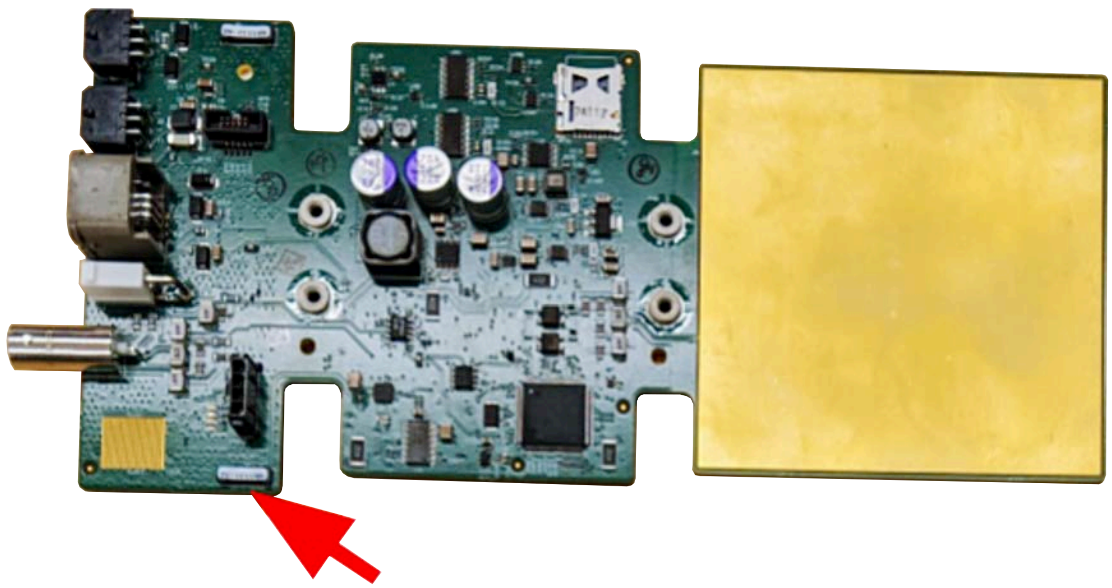


Figure 3-73 StormTrooper board with lighting detection IC and antenna.

This sensor turned out to be quite sensitive and we had to do significant tuning to reduce effects of nearby devices, such as the transponder and altitude control system. Unfortunately, the StormTrooper on the truss turned out to be too affected by the transponder to be useful. The board on the bottom plate of the balloon, however, did frequently pick up real storms, but did not give us any real new data compared to the corona discharge detector. Loon also found the built in estimates of intensity and distance provided by the IC to not be very trustworthy in the application.

An ***electric field detector*** was implemented in both locations as a parallel plate capacitor (roughly 3.5 x 3.5 inches with a 93 mil spacing- which was roughly 130pF) made out of copper planes on the PCB.

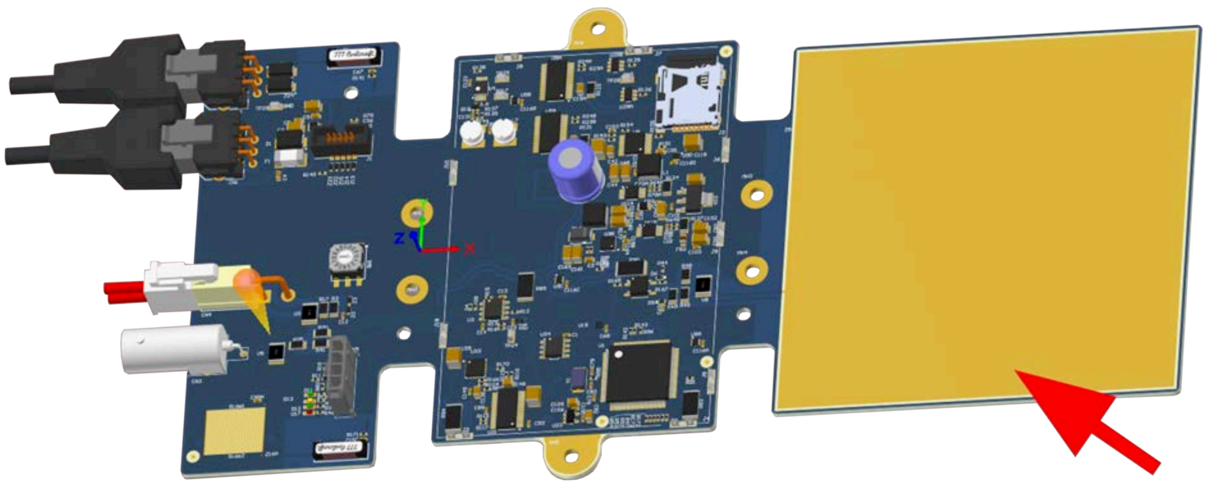


Figure 3-74 ***StormTrooper board.***
Electric field detector is shown on the right.

Unfortunately, this did not work as well as expected, and produced little valuable data. It was designed to be sensitive to extremely large rates of change of electric fields, but we suspect the majority of storm events were not large enough to be reliably captured here.

Most Common Electrical Failure Modes

As alluded to before, most storm induced failures occurred on or near the top plate of the balloon. Of these failures, a few stand out, and were reproducible, either in flight or on the ground with the lightning tester. One common failure was a power supply IC that was rated to 80V and operating on a 50V bus, protected with a 53V 1000W TVS diode. Unfortunately, currents induced on the CAN cable produced a large enough surge that, even with the TVS protection, the voltages exceeded the rating of this part and led to a failure. This was confirmed during lab testing with our lightning transient generator. Loon later switched this IC to a 100V rated part and was not able to reproduce the failure.

Another common issue was streamers forming from the temperature RTD sensors hanging inside of the balloon to measure the temperature of the helium. This could be observed as nonsensical temperature reading, and sometimes caused failure of the sensing circuit. We confirmed this failure mode by observing charred sensors during post-landing failure analysis.

Once balloons had landed, we also observed some scorching of the cable itself that runs to the Apex, likely due to streamers forming at pinholes in the insulation of the cable. We are unsure if this typically happens when the balloon descends or lands through a storm, or if it occurs while at float. In two cases, we found that these streamers had also melted holes in the balloon film that coincided with the melted locations on the cable.

Early on in the project, Loon went through an effort to decrease the weight of the top plate assembly which shifted the Apex board to be the highest point, as well as a number of wires exposed on the top of the balloon. We quickly found this to be more susceptible to lightning transients and actually experienced cases where squibs fired due to a transient. We were able to reproduce this effect with an on-ground lightning attachment survey.

In many other cases, the electronics did survive electrical transients and/or the flight vehicle survived turbulence, but the flight had to be landed in a matter of days because of leaking. Streamers from the apex and/or CAN cable are suspected to have created or enlarged holes in the envelope and/or ballonet in multiple flights. Note that helium has a lower dielectric constant than air, making it easier for streamers to form. There are examples in this class where both air and helium leaked, meaning that the ballonet was damaged (and probably the outer envelope as well).

Direct Balloon Impingement

Loon has at least one well-studied example (and other suspects) of a failure mode in which turbulence caused the balloon to tilt quickly enough that it was possible for the solar panels on the payload to come in contact with the balloon film. We expected this failure mode to be reduced in likelihood for the next generation platform due to the extra-long downconnection between the payload and balloon.

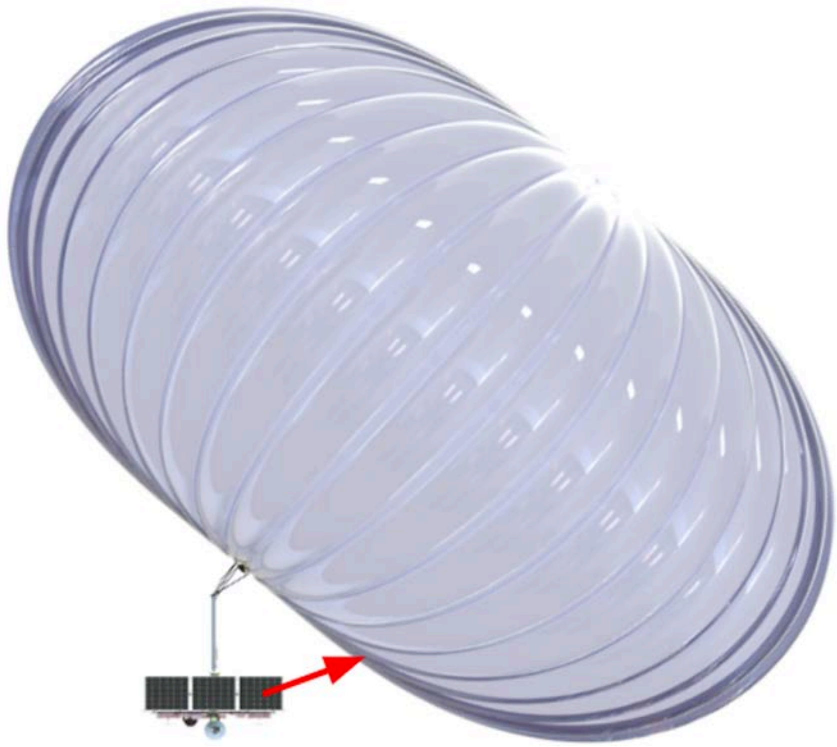


Figure 3-75 Balloon tilt leading to direct impingement.

Diagnostic and Verification Testing

Loon primarily used two types of on-ground testing to determine weak points in the design relating to storm-induced electrical transients.

The first of these tests was referred to as a lightning attachment survey. This test used a large tesla coil attached to the Apex CAN cable to simulate where transients would come up the cable and form streamers off of the top plate.

From this test, we roughly found that if you can roll a beach ball over the top plate and touch a piece of metal, you can probably get streamers forming off of it. In addition, anything that was more towards the edge or pointy had a higher likelihood of streamers forming.

The image below shows the revised top plate mentioned before with the Apex board on top, with streamers coming off the board itself. During this test, Loon was able to reproduce the squib firing, as it happened in flight.

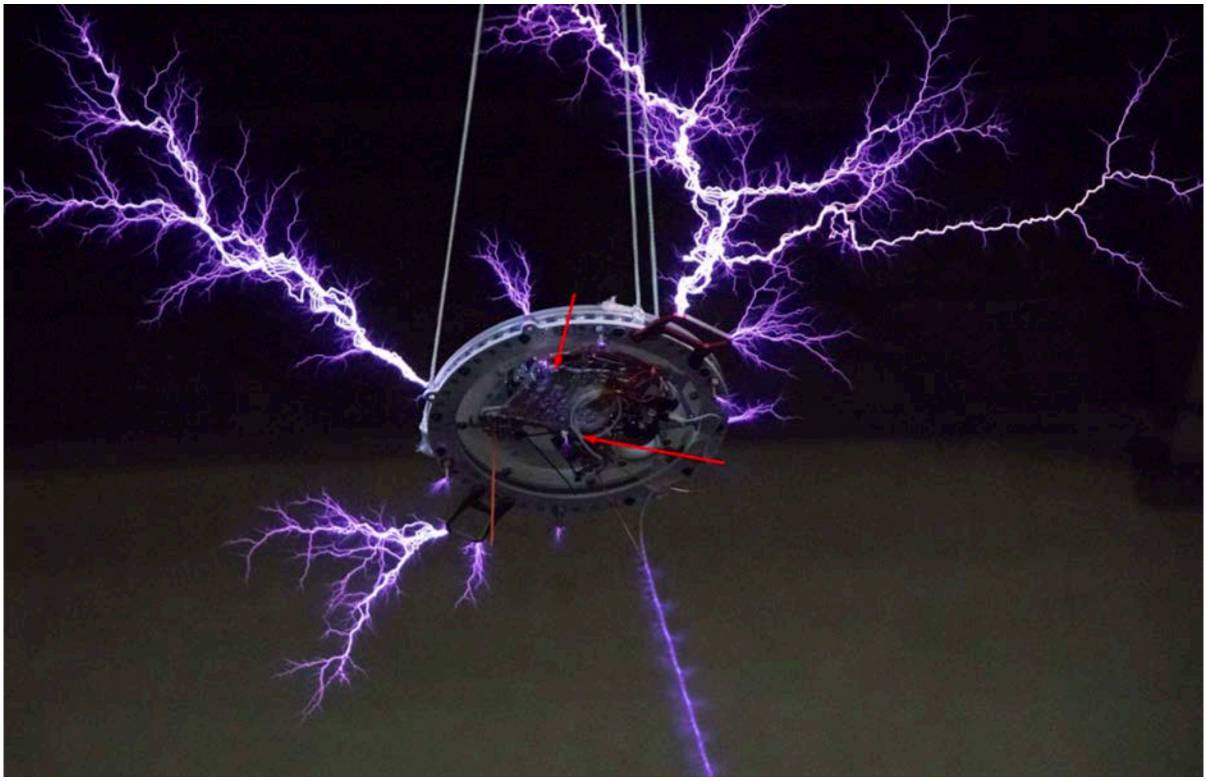


Figure 3-76 Apex assembly undergoing lightning attachment survey.

The other effect that was very visible was the presence of pinholes along the cable, which Loon believed could explain some of the damage witnessed on recovery to the cable and film.

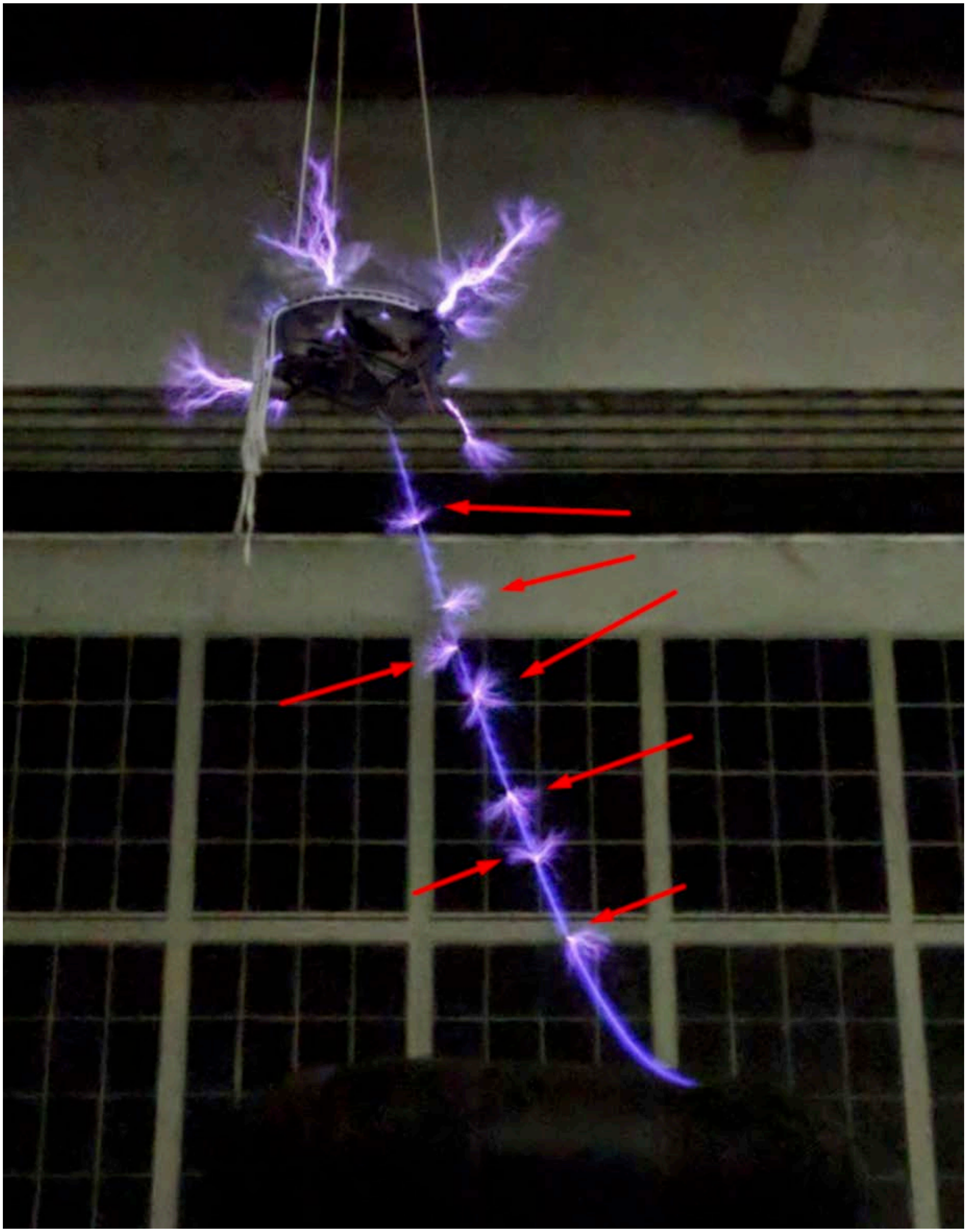


Figure 3-77 CAN cable pinholes during lightning attachment survey.

After initial diagnostic testing, Loon started adding a lightning cage over the entirety of the top plate to shield the electronics and squibs from these streamers by allowing the streamers to form off the cage instead. For any major revisions, we also repeated the test to determine if we missed any points in the design that may cause trouble. Overall, this was a very qualitative (but helpful) test, as typically failures weren't visible (system is unpowered), and images must be analyzed to look for any issues.

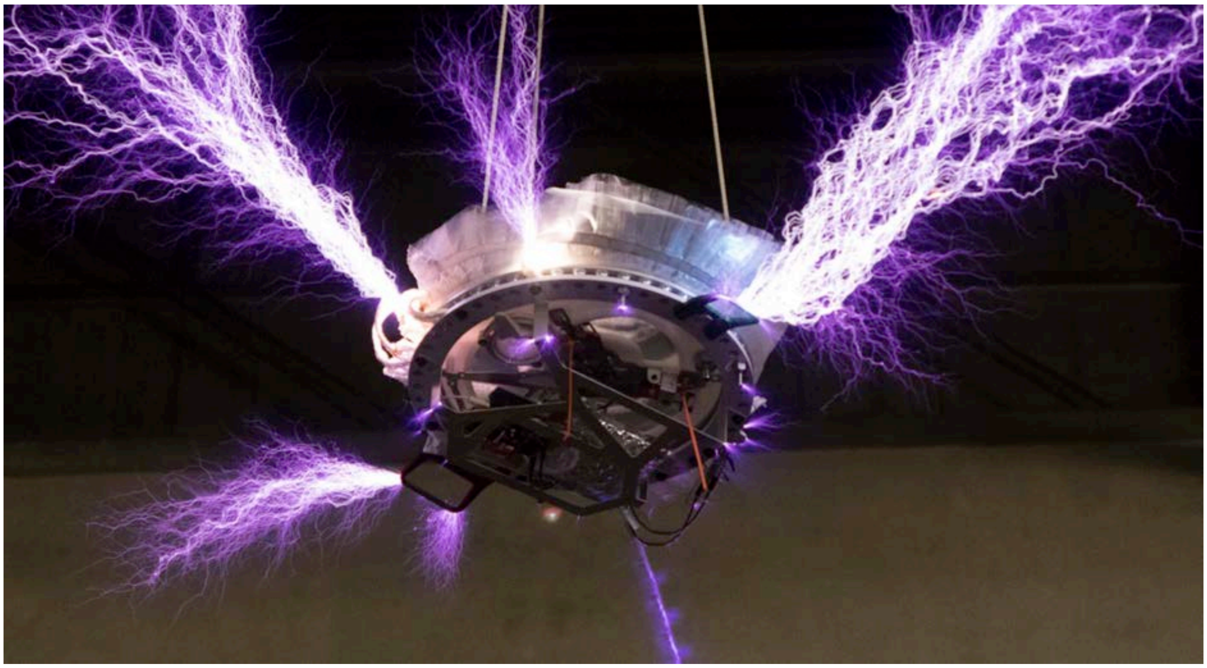


Figure 3-78 Lightning attachment survey of apex assembly with lightning cage installed.

For additional, more quantitative testing that was also easier to run, Loon purchased a transient generator that was capable of producing Level 3 indirect lightning strike waveforms defined in the DO-160G (Environmental Conditions and Test Procedures for Airborne Equipment). This allowed Loon to perform cable bundle and pin injection tests simulating what a board might experience.

In general, Loon found the pin injection tests to be able to better replicate failures seen in flight. Cable bundle injection worked as intended but tended to not cause system failures. After setting up this equipment, Loon generally tested all connectors that may be susceptible to external transients to the following waveforms:

- DO-160G, Waveform 4, pin injection
- Single-stroke

- Level 3 (300V)
- 10 strikes each polarity, 2 seconds between strike

This transient generator was designed for testing equipment that was installed inside the fuselage of an airplane. The Loon operating environment was simultaneously more harsh (no fuselage protecting the electronics) but also less harsh (lower frequency and intensity of direct lightning strikes at that altitude). The selection of limits was a balance between what level of hardening was achievable with a lighter-than-air vehicle and what was required based on operating environment and risk level.

Hardening Techniques

Adding a lightning cage around the top plate and other sensitive electronics was by far the most impactful change. If creating a similar vehicle, be sure to shield electronics as much as possible, and connect those shields to your system ground. Generally, Loon also ensured systems grounds are connected together as much as possible (such as shields or chassis and power, digital, and analog ground). The goal with all shielding was not to block or insulate subsystems, but rather to provide a low-impedance path for lightning-induced currents to easily flow around the subsystems.

For power protection components, Loon paid particular attention to the clamping voltage of TVS diodes and assumed that they would likely meet or exceed those slightly. For example, we used a TVS diode on most of the power rails with a working voltage rating of 53V, which suited the 50V max bus, and had a clamping voltage of 85V. We ensured that all components behind this protection were at least 100V rated, as we experienced enough energy in transients to cause an 80V part to fail. In general, we tried to use 1kW rated TVS diodes wherever possible (versus the standard 400W or 500W parts).

For digital connections that were sensitive to added capacitance (such as CAN) which large TVS diodes typically add, we went with thyristors. We found they provided sufficient protection while also not impacting signal integrity. For RF signals, we found that Gas Discharge Tubes (GDTs) can provide sufficient protection.

From a system architecture perspective, an additional improvement that was made was to add a backup battery to the top plate of the balloon. This battery simply consisted of a small, 3 cell pack but it allowed for about 6 hours of operation of the Apex to fire the FTS when needed. In addition, we moved the Inmarsat modem to the top of the balloon to provide a second communication channel. In the case of a failure of the cable between the balloon and the top plate, this allowed much more flexibility in where and when to terminate the balloon.

Hardware and Software Watchdogs

A significant challenge of deploying hardware to the stratosphere is there is no option for human interaction with the hardware or software once the balloon is launched, aside from the predefined interface over SATCOM for the full duration of the flight (300+ days). Therefore, it was essential for Loon to develop a number of strategies for making the boards more robust and self-sufficient. From early on in the project, Loon implemented the software watchdog feature on the STM32F4 microcontroller. This forced the firmware to pet the watchdog every second to make sure we didn't accidently get into a corner case and lock up the microcontroller. However, we noticed that, on occasion, peripheral devices (like devices connected over I^2C or SPI) locked up and caused a portion of the board functionality to degrade. In addition, there were occasions where we would see the MCU lock up and not recover, but power cycling the whole board would fix the issue. This was especially a problem for boards with local backup power where power couldn't be cycled remotely.

After trying to add power specifically switched for peripheral power domains, and some other changes to improve robustness, Loon decided to implement a hardware watchdog circuit that would automatically power-cycle the whole board if a pin wasn't toggled every few seconds. After implementing this on most of the new boards, we found the system to be quite robust, with increased stability. Loon also added latches and monitoring circuitry so that we could tell after the fact if a watchdog reset was observed.

Monitoring and Fault Reporting

Since the system must function without human intervention for 300+ days, designing in extra monitoring of the system is essential to understanding if a board is starting to fail or has failed, and what the root cause may be. In general, Loon tried to monitor voltage on all rails of the board, as well as current on most of the critical or more interesting rails. This allowed us to have a really good understanding of where energy was being used in the system, and if there was unexpected usage in edge cases.

In addition to power, Loon also monitored board temperature with small I2C based temperature sensors that allowed us to easily monitor 4 locations on the board, then run a control loop on the heater resistors (integrated on the board and coupled to the ground plane for best thermal distribution) to keep the board above its minimum temperature rating.

CAN networks typically have a few metrics built into drivers to understand warnings, errors, and bus-off conditions to assess the health of a particular network. Loon monitored these, but also strategically added voltage dividers with large low pass filters on the bus in select locations. This allowed us to monitor the state of the bus, as if there was a fault we could tell specifically what was going on with the physical wires (like say one wire shorted to ground or a power rail). It also gave a proxy for the amount of traffic on the network (the further apart CANH and CANL are, the more traffic).

Another very useful feature that Loon implemented was a fault monitoring and latching system. This allowed us to define a large number of fault conditions on a board (for example, voltage out of spec), then, if any of these occurred we would latch them on the vehicle side until the fleet management system on the ground acknowledged and cleared them. When we first implemented this, we found there were a lot of transient conditions that we did not catch in the typical, under-sampled telemetry that, when fixed, allowed for a much more robust hardware and software stack.

Reliability

Loon's targeted average mission duration of 300 days for a very novel communications platform meant that reliability was a critical factor in all elements of the vehicle. Since the vehicle could not be physically accessed once flying, a critical component outage would result in the need to end the mission and land a very expensive vehicle providing a telecommunications service. This focus resulted in a number of key lessons and recommendations for stratospheric operation:

Note: For a complete review of Loon's reliability requirements see the Loon Reliability Requirements Document (LnArt-Loon RRD).

A Design-For-Reliability (DFR) ethos was understood by Loon to provide significant benefits as the system architecture began to mature. Incorporating reliability early in the design cycle drastically reduces cost by reducing much more costly downstream reliability failures. Based on early learnings, we did try to build reliability early into the designs (cost of reliability increases non-linearly through the design cycle). We used this in later versions of the vehicle to greatly increase longevity (10x vehicle life despite 10x complexity).

A system reliability model and reliability budget were both essential for design targets, business planning, and flight operations.

- Established design specifications enabled good reliability metrics for critical components. These testable specifications greatly improved life estimation through design and test as well as inflight analysis. Note that industry standard tools go hand in hand with this as noted in the next section.
- Business planning accounted for lifetime reliability trends (new product introduction) and growth (maturation of builds and designs). It's essential that a predicted lifetime distribution (preferred over an average) based on reliability estimates or empirical data should inform business plans for operational cost. Similarly, reliability trends from prototypes to production sunset should be studied and modeled continuously on a subsystem basis rolling up to a running reliability budget (example–longevity of new slip ring materials, or introduction of new ACS) for an up-to-date aggregate system estimate for business cost purposes on at least a quarterly basis.
- Flight operations worked within cross-functionally acceptable reliability parameters. Operational usage outside of these recommendations was de-risked in preparation for failures with pre-agreed procedures or CONOPs. Critical component and system level prognostics should be pursued to enable predictable operation of the vehicle.

Loon used industry standard tools and methods to understand the design from a reliability perspective as soon as possible in the design cycle using these assessment tools:

- Reliability system, subsystem and component modeling (example: bearing lifetime analysis and estimated descent system reliability). High confidence intervals for safety critical system reliability are encouraged for new systems (versus confidence growth via testing or flights) for risk purposes.
- FEA reliability estimation via rainflow counting/fatigue, thermal stress analysis (eventually using Sherlock).
- DFMEA/DRBTR methods and analysis were used for risk classification and as design mitigation tools. Using these reliability engineering tools to drive design concepts ahead of CAD eventually became the goal.

- HALT testing identified weak spots in the design
- Replication of known in-field failures was essential to understanding the root cause for many Loon systems. This often required unique efforts at reproduction in the lab to mimic the combination of stresses causing these issues (See details on "Despin Rotating Actuator" on page 103).

Loon Reliability Concept Recommendations

A detailed failure analysis plan for critical components should be synthesized as early as possible in the design cycle (easy example could be the Five Whys method) to extricate possible causes for unknown failure modes (example–inflight compressor oscillations were actually captured via audio from a GoPro test flight video). The end result of this template should be presentable as a complete document from failure to root cause. Part of this analysis prep work should include thorough prep, pre-launch and launch imagery for investigating issues and ruling out certain root causes.

End-of-line testing should reflect component criticality and feed into the design cycle (example: highly critical components go through HASS with failure modes and life usage understood). Some systems had insufficient test coverage in comparison to others which resulted in early life failures.

Low sample size reliability evaluation is valuable and sometimes necessary (example: HALT, high cost system reliability, complex subsystems) but implications for scale (failure mode distribution and likelihood) may be unknown; assess this risk in the context of a system FMEA and consider addressing this in the greater reliability plan.

A well-equipped reliability and failure analysis lab is essential. Ongoing reliability test (ORT) programs and a well-equipped analysis lab complement each other in the design-reliability feedback loop. ORT should be considered de facto for critical components. Recovery and landing damage impact failure analysis; procedures should be developed for retrieving and handling samples.

Loon Reliability Lab Setup Recommendations

Based on the electrical and mechanical failures commonly encountered, below could be a Loon-centric Reliability Testing and Failure Analysis Lab essentials list.

- Testing Lab:
  - » Pressure Varying Environmental Chambers
  - » HALT Chambers
  - » Mechanical Force-Extension Testing Capability
  - » High Delta Testing (temperature and time) Capability
- Failure Analysis Equipment:
  - » High quality microscope
  - » Optical comparator or CMM
  - » 2D X-Ray Machine
  - » Mission Appropriate Electronics Test (These are absolute minimums-consider the DO-160 testing discussion in this document for storm considerations)
    - ESD
    - LCR/TDR Devices
    - GTEM and Spectrum Analyzer
- Outside Lab Access and Expertise[1]
  - » Mechanical Cross Sectioning Capability
    - SEM Capabilities (EDS, FIB)
  - » TGA/DSC (Thermo-Gravimetric Testing / Differential Scanning Calorimetry)
  - » CT Scanning (2D and 3D)

1 Strongly consider a dedicated lab manager for instrumentation, metrology, and test coordination

Loon-Unique Stresses and Reliability Considerations

Some examples of Loon-specific stresses which made reliability challenging included:

- Large temperature deltas
  - Caused unusually low cycle thermal fatigue damage
  - Caused issues at material interfaces from CTE mismatches
- Cold temperature soaks
  - Solder experienced embrittlement; characteristics of eutectic point shifted for fatigue and brittle failure modes (especially true for large PCB footprint wave soldered devices such as ceramic filters).
  - Lubrication state changes (see "Bearings and Lubrication" on page 152).
- Low pressure and solar/DWIR effects (both worlds)
  - Required consideration of both radiation and convection coefficients, leading to very location specific solutions (example, systems in slipstream of lateral propulsion versus those on top of balloon facing space).
  - Arcing can occur over larger pin distances (Paschen's law) compared to the same voltages at higher pressure locations on the ground.
- Electrical and Solar Aging Effects (discussed at length in the balloon and avionics section, but consider long term UV degradation and its mechanical impact as well)
- Vibration and Shock
  - Vibration transmissibility of the system is important. Loon specifically experienced this with the altitude control compressor and lateral propulsion systems, impacting electronics and connectors. Communication platforms that required delicate positioning (B2X connectivity) needed to consider both the short term performance and longevity impact.
  - Mechanical shock, long term vibration, and cable entanglement were all key areas and Loon needed to be cognizant of high impulse short duration conditions such as launch or parachute deployment, as well as high cycle, low stress ones (cable fatigue or connector fretting) in electrical and mechanical connections.

Chapter 4

Communications Systems and Service Design

Loon's vision to provide internet access from balloons could be accomplished in several ways. This chapter describes the overall architecture and the underlying hardware and software systems used by Loon during its development phase and for commercial deployment to bring the internet up to the balloons and back down to the users.

Communications Architecture and Evolution

One approach to providing internet access from the stratosphere is to use a fixed (non-mobile) terminal that connects wirelessly to the balloon. That terminal can then deliver internet access via Wi-Fi to a group of users nearby (typically a household, school, or small village center). Loon showed that this approach was viable after conducting initial proof-of-concept tests using a terminal mounted at the test users' sites. These tests used Wi-Fi radios to connect from the balloon to the terminal, but other point-to-multipoint wireless technologies could also be used for such Fixed Wireless Access services. Loon did not pursue this Fixed Wireless approach due to the cost and need to install this special terminal, both of which are barriers to broad adoption.

Figure 4-1 ***Early Loon fixed wireless service.***
Installing terminals on roof of school in Agua Fria, Brazil.

Loon instead settled on the “direct-to-handset” approach using LTE, the same mobile/cellular network technology used globally in terrestrial mobile data networks. The two primary benefits were:

- The served population was able to use Loon with any off-the-shelf LTE phone, including low-end feature phones.
- LTE was designed for a similar use case: the base station was far away from users, and there were hundreds to thousands of users per base station, making LTE capable of reaching the user and meeting the throughput requirements of Loon.

Figure 4-2 ***Loon direct-to-mobile handset service launches in Kenya.***
At service launch event in early 2020.

The overall Loon network architecture is shown in Figure 4-3. As with other mobile networks, there are three primary network layers:

- ***Access Layer.*** How users *access* the network. For a typical ISP or broadband network, the Access Layer consists of the fiber, phone, or coaxial cables to the user's home or facility. The router or the terminal provides an Ethernet or Wi-Fi connection for the users. For a mobile network, the access layer consists of just the last hop, from the Mobile Network Operator's (MNO's) base stations to the user's phone.
- ***Backhaul Layer.*** Everything between the base stations and the core network, including the cables or fibers and any other routers or other equipment in that path.

- ***Core Layer.*** Racks of electronics providing various routing, packet processing, authentication, and accounting services, including the connection to the internet and the telephone network, typically found in an MNO's central office facility.

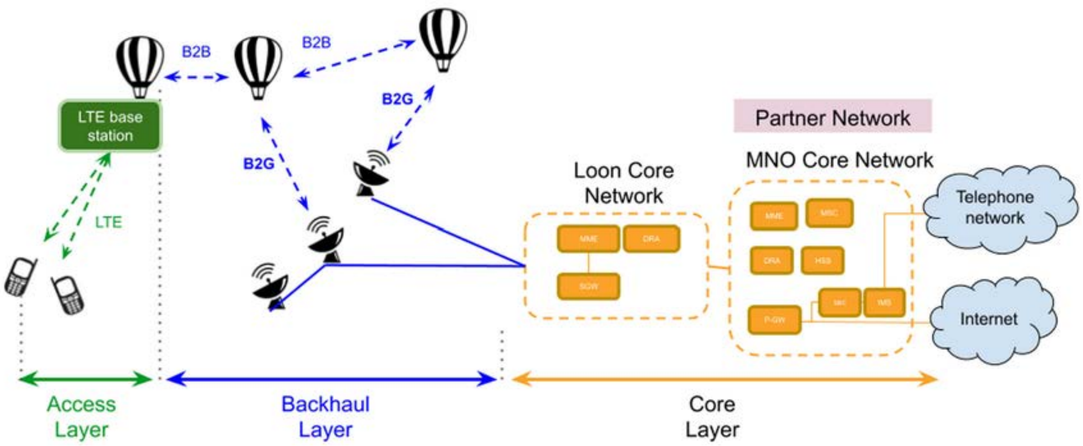


Figure 4-3 Loon LTE network architecture: Core, Backhaul and Access layers.

Loon's network differs from every other LTE network in two key areas:

- Loon's LTE base stations aren't mounted permanently to towers or buildings but are floating on balloons, 20 km up in the stratosphere and constantly moving across countries, continents and oceans.
- Loon's backhaul to each LTE base station is not a fiber line from the operator's central office to the base station. Instead, it consists of several of those fiber lines going to Loon ground station sites across the service area, each of which is tracking the floating balloons. These ground stations deliver the backhaul signal to balloons, which then relay it from balloon to balloon until it reaches each LTE base station in the sky.

As can be seen in Figure 4-3, Loon's backhaul network consisted of balloon-to-balloon links (B2B), balloon-to-ground links (B2G), ground stations, and fiber links back to Loon's core network. For the B2B and B2G connections, Loon explored:

- Free-Space Optical Communications (FSOC) using lasers
- High-speed microwave point-to-point radio using either the 71 GHz–86 GHz band (E-band) or the 39 GHz band

Wireless radios became the preferred alternative for the B2G portion due to the higher FSOC cost and FSOC's inability to pass through clouds. E-band was chosen over 39 GHz due to the amount of bandwidth available. Once the choice was made to use E-band radios for balloon-to-ground links, it was also the best choice for B2B connections, which could use that same radio system. Because the system on the balloon is used for B2G and B2B, it is named B2X.

A balloon with two or more B2X systems could be used to receive backhaul from the ground and extend it to other balloons that could not reach a ground station. Loon chose to include three such B2X systems on each balloon, which allows the creation of redundant meshes.

For the past few years, Loon's fleet was regularly forming meshes of dozens of balloons connected to several ground stations. Total link distances extended across thousands of kilometers, sometimes stretching across South America and almost across Africa.

The following figures show a variety of Loon backhaul meshes in actual operation.

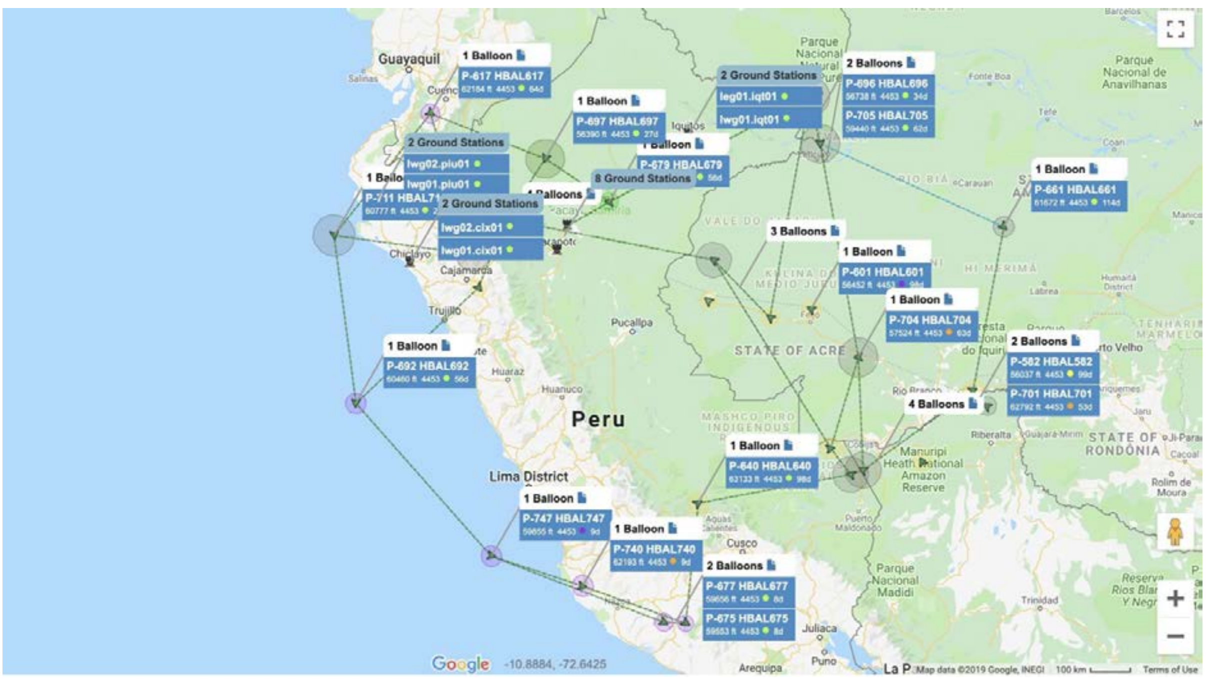


Figure 4-4 ***Mesh in northwest South America.***
28 balloons with 14 ground stations (four sites).

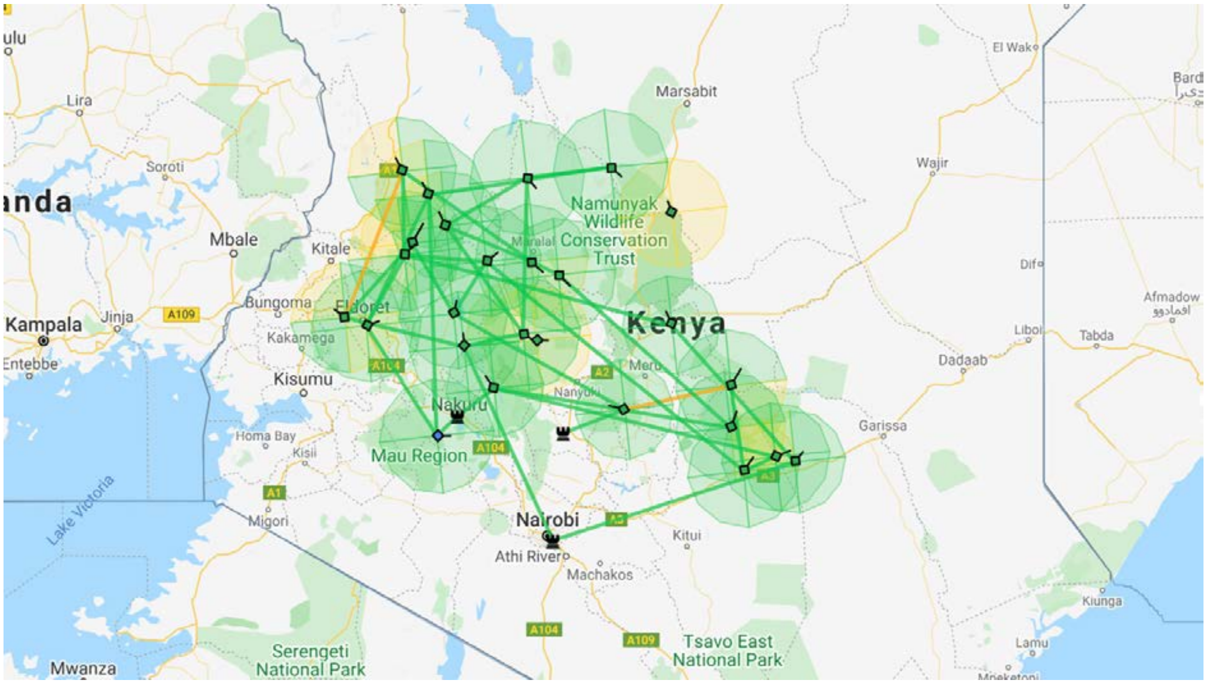


Figure 4-5 ***Dense mesh in Kenya.***
25 balloons with six ground stations (three sites).

Figure 4-6 ***Mesh in South America.***
25 balloons with early software and hardware.

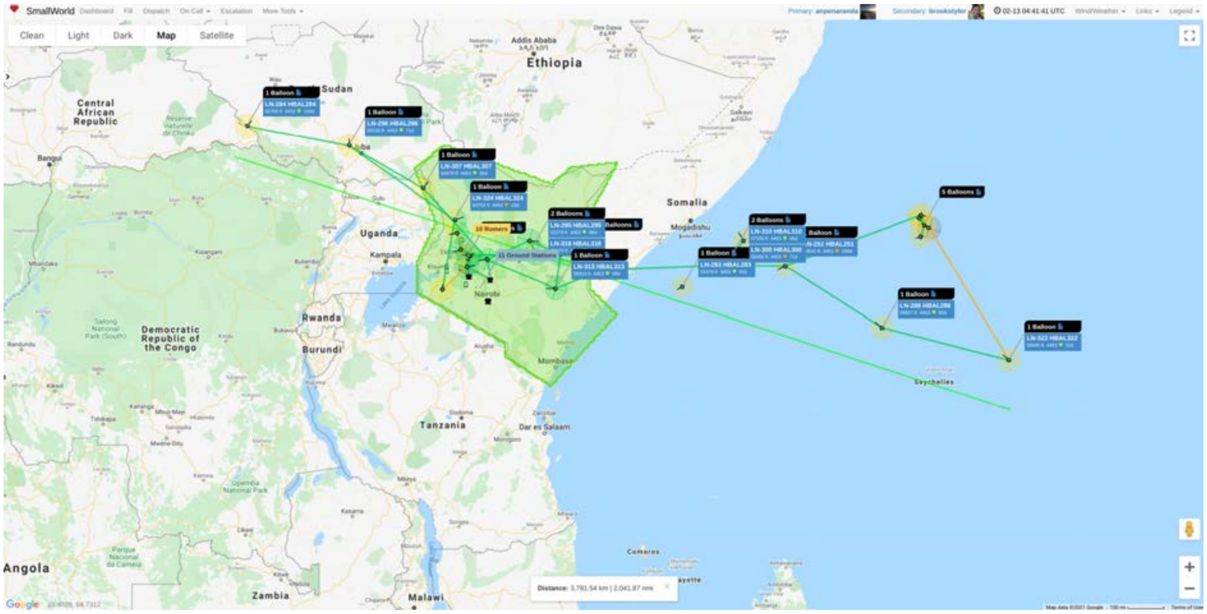


Figure 4-7 ***Long mesh from the middle of Africa to the middle of the Indian Ocean.*** Extending almost 4,000 km with almost 40 links totaling more than 6,000 km.

Service Payload Architecture

Each Loon flight vehicle had several systems that made up the Service Payload. These elements were responsible for delivering the actual service (LTE internet) to the user. The “payload” indicates that these systems were not required for the operation of the flight vehicle itself. Potentially, an entirely different service payload could be installed on and operated from the same balloon.

The Service Payload subsystems were:

- ***Comms Node.*** A central Ethernet switch and packet processor for interfacing with the remaining service payload systems, also supplies, controls, and monitors power to the B2X and LTE systems.
- ***One single-sector or two dual-sector LTE base stations.*** Each connected via Ethernet to the Comms Node.
- ***Three B2X subsystems.*** Each connected via Ethernet to the Comms Node.

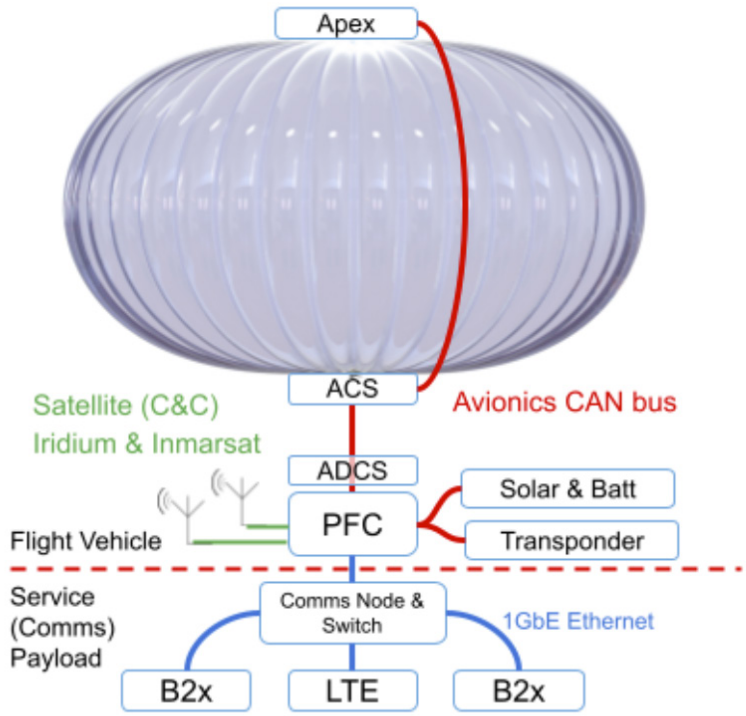


Figure 4-8 ***Vehicle internal communications architecture.*** Avionics and flight-related systems above, and Communications Service Payload below.

The B2X systems could transport almost 1 Gbps each. This high link capacity meant that a dozen or more balloons could be backhauled through a single B2B or B2G link even if that link had reduced capacity due to weather or distance. The backhaul network management software (see "Minkowski Drains" on page 314) orchestrated all the B2X and ground station nodes to create network topologies with redundant paths and to spread the traffic across as many ground stations as possible. Still, bottlenecks could always occur for some arrangements of balloons.

Communications Node (Comms Node)

At the heart of the Service Payload was the Communications Node or Comms Node, a central Ethernet switch, packet processor, and computer to interface with the remaining service payload systems. The Comms Node provided these functions:

- Centralized networking functionality (common to all service payload systems)
- Power distribution control and monitoring to all service payload systems (LTE, B2X, etc.)
- ADS-B receiver used for determining, in real-time, the presence, location, and speed of other Loon balloons in the area
- Interface to the core balloon avionics (PFC, or Primary Flight Computer), responsible for satellite communications back to Loon's Fleet Management Systems software and Minkowski/TS-SDN, enabling control and monitoring of the Comms Node and B2X nodes, through SatCom.
- Control of the Wi-Fi backup backhaul electronics and antenna.

The additional service payload elements are described in the Access and Backhaul sections below.

Access Network Layer

Loon selected LTE as its access-layer technology, that is, as the final link from the balloon to the user. Any standard LTE phone from dozens of major and hundreds of small providers could be used. LTE was just starting to be adopted in Loon's target market as Loon was ramping toward commercial deployment. By the time Loon had commercial agreements in place, a reasonable portion of the population in our planned service territories had LTE-enabled smartphones or feature phones. Some phones were as inexpensive as US$60-$80 and heading to below US$50, enabling even wider adoption.

An issue with these low-priced phones is that their LTE radio performance is significantly worse than the phones Loon used during the development and testing of its LTE subsystems. This performance readjusted Loon's expectations for the number of users that could be serviced, the number of GBs that could be delivered, and from which environments the users could successfully connect to Loon.

LTE Architecture

When most people think of LTE, they usually think of it as just the radio used by the base stations to communicate with the phones. However, the LTE system extends back to the mobile network operator's facility. An LTE Core Network, called an Evolved Packet Core (EPC), manages the LTE network and the user's connection to it. The EPC connects to each LTE base station, called Evolved NodeB, or eNodeB (or often shortened to eNB), through a backhaul connection, typically a series of fiber, Ethernet and wireless connections, or, in very remote locations, satellite connections.

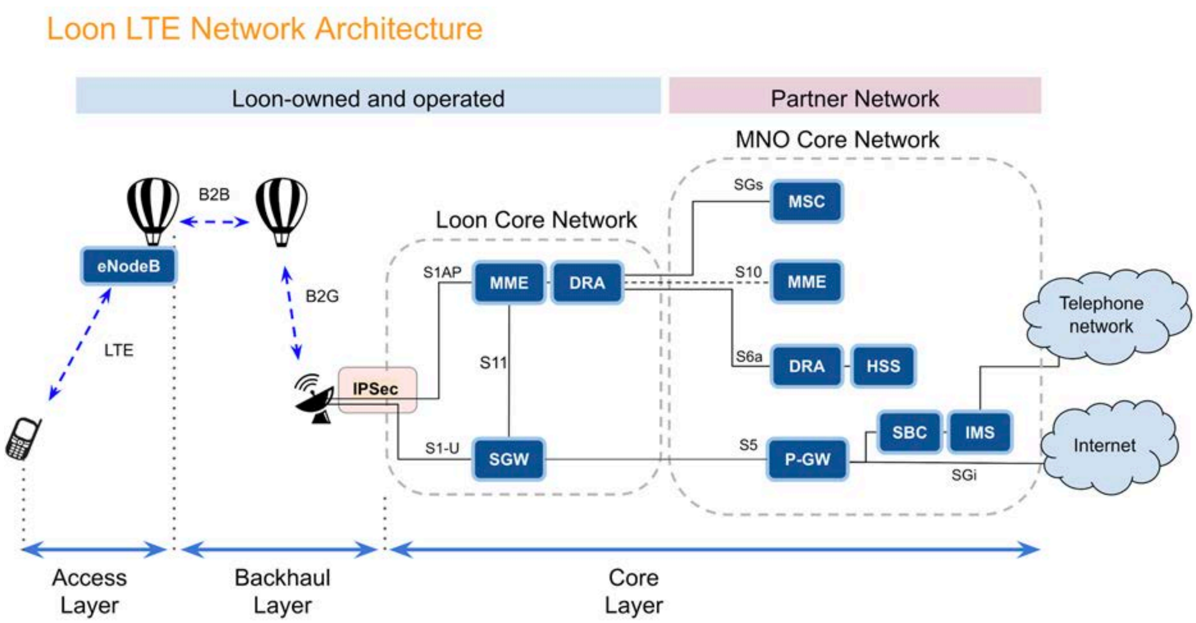


Figure 4-9 LTE network architecture: From core (EPC) out to base station (eNodeB).

LTE Performance

As described in section "Service Payload Characteristics and Impact on Cost and Service Performance" on page 40, the critical service metrics of interest to Loon are:

- Data rate or throughput
- Availability

Due to the nature of Loon's sensitivity to winds and weather, availability is most impacted by whether or not balloons are in the service area. Even if a balloon covers the area, users may not be getting service because the capacity of the sector covering them is too low to accommodate all the demand in that area or because the user is indoors or in a car, too far from the balloon's location.

The two design aspects of an LTE system that most affect capacity and performance are the number of sectors and the bandwidth per sector (e.g., 10 MHz vs. 20 MHz). However, several other factors are essential, including interference between Loon's own sectors or between Loon's sectors and terrestrial sectors belonging to the same partner operator or a different operator using the same channel.

Interference can be decreased in several ways:

- *Move the sectors away from each other:* but this also leaves coverage gaps between those sectors.
- *Use different radio channels for the interfering sectors:* This would reduce interference, but it would require two or three times as much bandwidth for the network. This spectrum would cost the operator two to three times as much (sometimes many millions of dollars) and was not generally an option for Loon.
- *Increase the sharpness of the antenna cutoff:* Theoretically, you can design the antenna to provide a stronger signal out to a certain angle and then very quickly decrease the signal level beyond that. But unfortunately, that is exceedingly difficult in practice and requires a bigger, heavier, and usually more expensive antenna.

Beyond these, many other detailed factors impact the actual in-field performance of a sector, including:

- Other antenna and sector performance factors:
    - » The characteristics of the antenna, particularly the *antenna "gain,"* affect the signal level and so can affect data throughput.
    - » The *number of antennas per sector* can be increased. For example, Loon used two, but many LTE base stations use four, which, in a typical terrestrial environment, often allows users to receive a higher data rate by supporting more "MIMO stream." In Loon's case, because of the distance to the user and the angle between the balloon and the user, the additional performance gain would be minimal, though the weight and power costs would be substantial.

- Aspects of the users and their phones:
  - What *type of phone* is being used: Low-end phones often have significantly worse performance than high-end phones. These low-quality phones have a major impact on how much data those users can download or upload and even where they can use their phone (in a building, a car, or only in an open field).
  - *How far the users are from the balloon*: Users directly below the balloon are 20 km away, and the farther they are from that position, the weaker the signal they will receive, in general. So, for example, Loon's LTE-Gen2 system (discussed below) offers peak performance about 10 km away from the balloon's ground position, and beyond that, the signal weakens.
  - The *user's environment and surroundings*: The Loon LTE signal level will be much lower indoors, so again, those users will get much lower data rates and may lose service altogether (lower availability). Likewise, if outdoor users have a building, trees, or a hill between them and the balloon, THEIR performance will suffer. Although Loon is at 20 km altitude and outdoor users directly below the balloon experience a clear signal, users only need to be about 30 km away from that spot before ground clutter starts to have an impact.
  - *What direction users are facing*: Even the user's own body can impact the signal level, so if the user happens to be holding the phone towards the balloon, this will work better than if the user's body or head is in the line of the signal.

The differences between phone models were not adequately considered in Loon's earlier stages. Still, after analyzing the performance of several phone models representative of those used in Loon's target markets, it appeared that many of those models had a performance that was 6 to 11 dB worse than those Loon had used to model and forecast its performance. For reference, a decrease in 6 dB means that the phone requires a signal level four times as strong to perform, and 11 dB requires a signal more than 12 times as strong. Thus, with many such phones connected to each sector, Loon's overall performance was much lower than planned.

Beyond reducing the data rate such phones would experience, this performance had an enormous impact on where those users would be able to connect to Loon. The signal level received while in a car is much lower and being indoors in a wood-framed single-floor residence is worse than the car. Moving into a multi-story cinderblock or other construction can completely block the signal even from nearby terrestrial towers. Loon was explicitly not supporting such buildings but did want to cover users in cars or near windows in most buildings expected in the target markets. By taking 6-to-11 dB off the signal level, users with those phones that could have received Loon's signal from inside their cars with a higher-end phone would need to be outside with a clear line-of-sight to the balloon. Likewise, because the users farther away from the balloon would receive lower signals, those with the low-end phones would need to be much closer to the balloons to receive service. This scenario reduced the service footprint of the balloon significantly.

Figure 4-10 shows the different coverage zones for the later Loon vehicles using the dual-Rickenbacker (LTE-Gen2) system. In red is shown the low-end phone outdoor coverage area, which extends out to about 30 km from a point directly below the balloon. Outdoor users with high-end phones are covered throughout the green area, getting out to about 60 km from the balloon's foot: more than four times the coverage area. Even at the peak signal location of this system (white spots), low-end phones can only barely connect from within a car. In contrast, high-end users in cars can connect across an area similar to the red area shown. Note that individual users, even with the same phones, will be experiencing a wide range of signal levels depending on how they're holding the phone, what direction they're facing, whether there are reflective buildings nearby, and other variables. These coverage patterns assume a fixed value for those variable factors. Therefore, the edges of these patterns should be interpreted as being quite broad and representing more of a substantial drop in the probability of service rather than an abrupt cutoff.

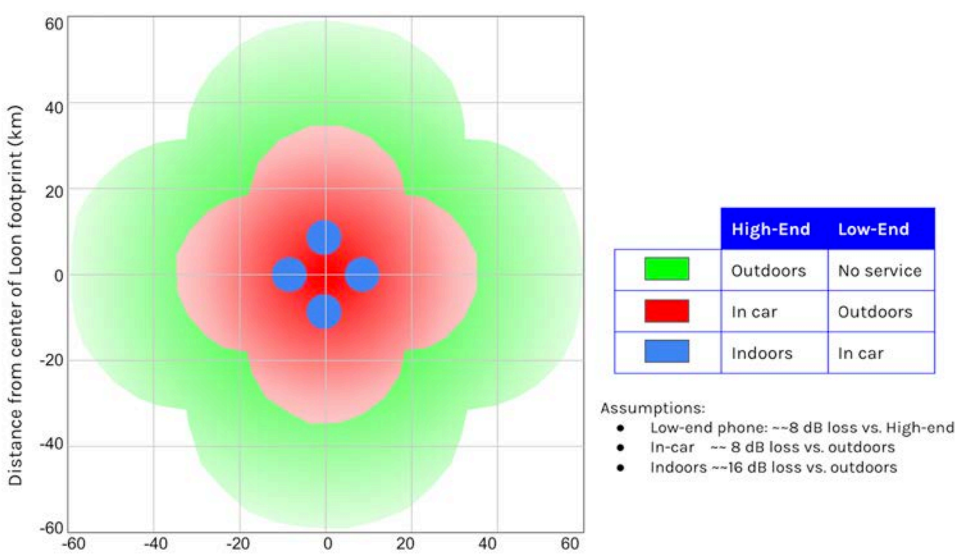


Figure 4-10 ***Rickenbacker coverage pattern.***
High-end vs. low-end phones and outdoors vs. in-car vs. indoor all affect coverage.

Certain settings that the partner operator may use in their existing 3G and LTE networks could also significantly impact Loon's coverage and the capacity it delivered. The settings are intended to prevent users' phones from being handed off from one sector to a lower-performing sector. The network is configured to prevent users from transitioning from a 3G or 2G tower to an LTE tower unless the signal level of the LTE tower is sufficiently high that the user will see at least the same level of performance they're getting on 3G.

Because Loon's objective was principally to serve users in areas with no existing mobile data connections, this effect was not as much of a concern.

LTE Subsystems

The Service Payload's LTE system is essentially a full LTE base station (eNodeB), similar in architecture and capabilities to a typical LTE base station on a tower. However, due to the unique operating requirements in the stratospheric environment, Loon engineers designed the base station hardware to support the necessary thermal management and ability to withstand the near vacuum, affecting several standard radio components.

Running on this Loon hardware was an LTE base station software stack from Nokia. This software was very similar to that running on Nokia's commercial LTE eNodeB base stations sold to mobile networks worldwide. The critical difference was that Nokia worked with Loon to customize it to support the unique requirements of a stratospheric vehicle with users that are at least 20 km and typically 40 km away. Using such a commercial-ready software stack was necessary to meet Loon's capability, performance, and robustness needs.

Loon began development with a single sector system ("Fodera") but upgraded to a four-sector system ("Rickenbacker") as Loon approached commercial service. In addition, subsequent larger vehicles were expected to require a substantially more capable LTE system, which was being called the Future Access System (FAS).

LTE-Gen1: Single Sector "Fodera"

Loon's initial LTE design had a single LTE sector, which had no more capacity than what is found on a single sector small cell tower. This single sector was projected as a single circular pattern on the ground. It was initially designed to cover as much area as possible, but this resulted in the signal level being relatively low for most users, requiring them to be outdoors and avoiding obstructions.

Nonetheless, Loon successfully used this system as the first to serve LTE users from the stratosphere, ever, by anyone. It was also used to provide emergency telecommunications services in 2017 during both the Peru floods and Hurricane Maria in Puerto Rico, where more than 200,000 people were served.

Antenna/Coverage Pattern

Several different antennas were explored on Fodera to provide a broad coverage pattern without a strong peak. With a typical sector antenna, the users that are directly in line with the aim of the antenna (at "boresight") receive the strongest signals because that is where the "peak gain" of the antenna is pointing. In the Loon case, because the antenna was pointed straight down, those users were already closest to the antenna, so they did not need the peak power. The alternate antennas pushed some of that energy away from the boresight direction to increase the signal level seen by users 20 to 40 kilometers away from the spot directly below the balloon.

This pattern allowed the system to cover as much as 5,000 square kilometers of area, far exceeding that of a typical single terrestrial sector, typically covering at most 500 square kilometers even in rural, tall-tower cases. Figure 4-12 shows the pattern over the San Francisco Bay Area to get a sense of this size. However, given the large area covered, this design significantly limited the density of users that could be supported. As a result, it was only viable over regions with a sparse population or one in which the population had a very low adoption of LTE phones. Fortunately, that was typically the case for Loon's target markets when Loon was testing its technology.

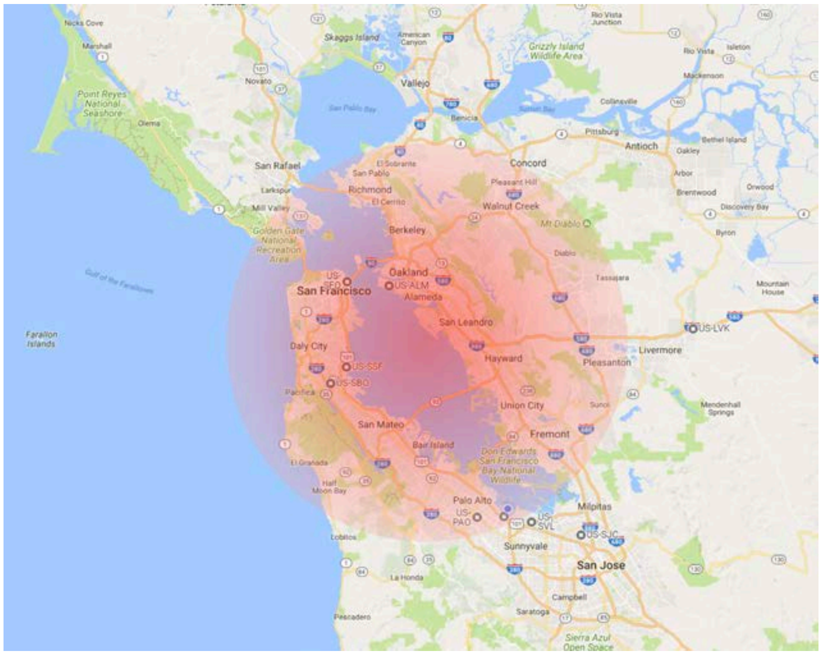


Figure 4-11 ***LTE-Gen1 Fodera sector simulated over SF Bay Area.***
Initial target for Loon service were areas with very low population density.

We chose to have a separate transmit and receive antenna, separated by almost two meters, enabled due to the large Loon bus structure. This scenario is very unusual for base stations, which typically use the same antenna for transmitting and receiving. The separation helped minimize interference between transmitting and receiving (tx and rx) signals and simplifying the electronics design's filtering and duplexing requirements.

The electronics architecture of the Fodera single-sector system was similar to a commercial small cell (small cellular base station). It was integrated with a base-band and radio transceiver on one board (codenamed "Wooten") and high-power and RF circuitry on two separate boards: a receiver (codenamed Rex) and a transmitter (codenamed Tex), each mounted just behind their respective (separate) antennas. Each antenna was dual polarity, so the entire system was "2T2R" (two transmit signals and two receive signals). This architecture enables faster throughput, though, due to the very steep angle-of-arrival of the signal at the user's location, Loon did not see as much MIMO (multiple-in, multiple-out) activity as would be typical for a small cellular base station mounted near the user.

LTE-Gen2: Four Sector–"dual-Rickenbacker": Two 2-Sector Units

The limitations of the Fodera system soon became a bottleneck, limiting the number of users Loon could serve, even in relatively low-density areas such as the Peruvian Amazon. As a result, Loon moved to the dual-Rickenbacker four-sector system for its initial commercial deployments to solve the capacity problems and allow each balloon to gather more revenue by serving more data. Initially, the solution was a single assembly, codenamed "Fryer Basket," with four antennas, each very similar to those on the Fodera (LTE-Gen1) single-sector system, but, because they were aimed midway between the horizon and straight down, each covered a large quadrant of the balloon's footprint.

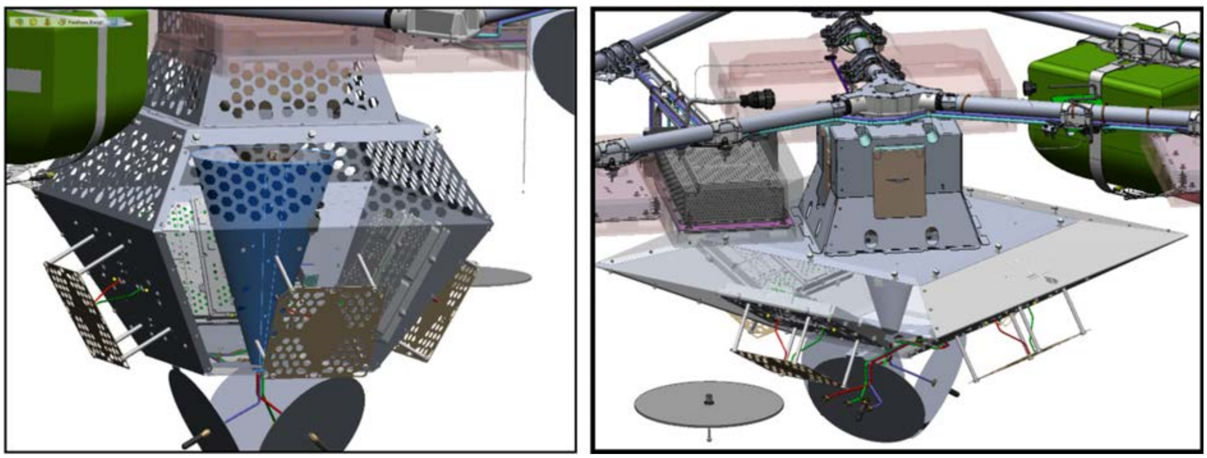


Figure 4-12 ***Early LTE-Gen2 designs.***
Fryer Basket configurations — far beam with antennas pointing 60° above nadir (directly down) and close beam with antennas pointing 30° above nadir.

As shown in Figure 4-12, two different antenna angles were proposed: 60° above *nadir* (which is the term for *directly down*) and 30° above nadir. The higher angle pushed the center of the sector far from the balloon while the lower angle kept the sectors a little closer to the balloon. However, neither of these was sufficient because the original antenna had far too broad a pattern., This meant that it could not deliver a decent signal level to the users when pointed out too far from the balloon, but trying to solve that by bringing the sectors closer to the balloon caused too much interference between sectors. The decision was then made to move to a higher antenna directivity (more focused pattern) for each of the four sectors to increase the signal level and reduce the interference.

The design then evolved into a dual-unit approach, with two identical Rickenbacker systems, each with two sectors. Each of the sectors had an antenna pointed about 30° above nadir and each of those had about 6 dB better gain than the original antenna. This new approach was accomplished simply by using a 2x2 arrangement of four antennas, each very similar to the original Fodera antennas. Because each sector spread the energy over a smaller area than the single sector of Fodera, the signal level was stronger, enabling a more solid connection with the phones. The improvements allowed more users to connect to Loon from difficult areas such as inside a car or to attain higher data rates than could be achieved with Fodera.

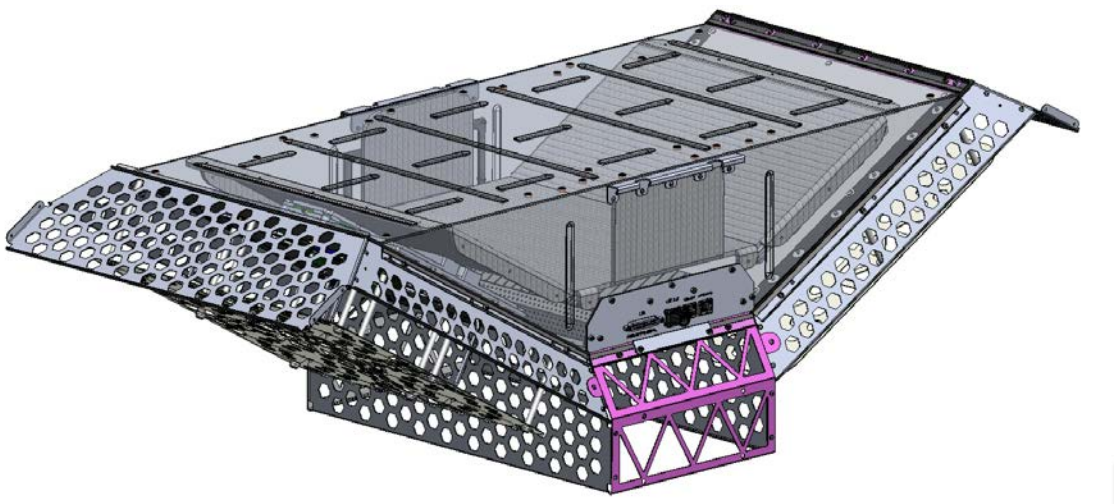


Figure 4-13 Single Rickenbacker two-sector LTE system.

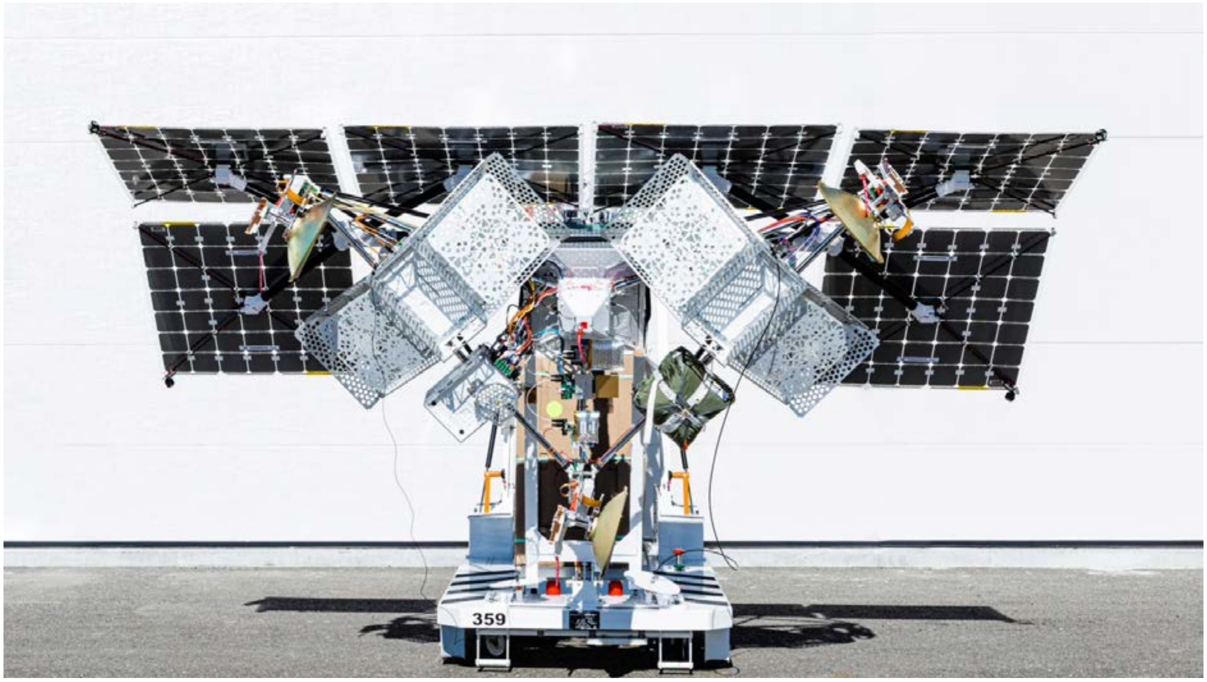


Figure 4-14 ***v1.6 Fully assembled and ready to fly.***
v1.6 system with dual Rickenbacker (LTE Gen-2) systems mounted on a launch cart.

Antenna / Coverage Pattern

The dual-Rickenbacker antenna pattern is shown in Figure 4-16. It depicts the SINR (signal to interference+noise ratio), which essentially indicates the usable signal level seen by users at those locations. The "interference" component assumes that all four sectors operate on the same channel, which is how these sectors were typically configured. Because they all have the same transmit power, users standing along the diagonals between sectors will see an SINR of 0 dB at best. This means that the signal level from the sector they are connected to will be equal to the interference level from the neighboring sector. However, this does not prevent them from attaching or transferring data because an SINR of 0 dB does permit some data transfer (see Shannon-Hartley theorem[1]) and because LTE adds additional mechanisms to increase that data rate.

1 https://en.wikipedia.org/wiki/Shannon%E2%80%93Hartley_theorem

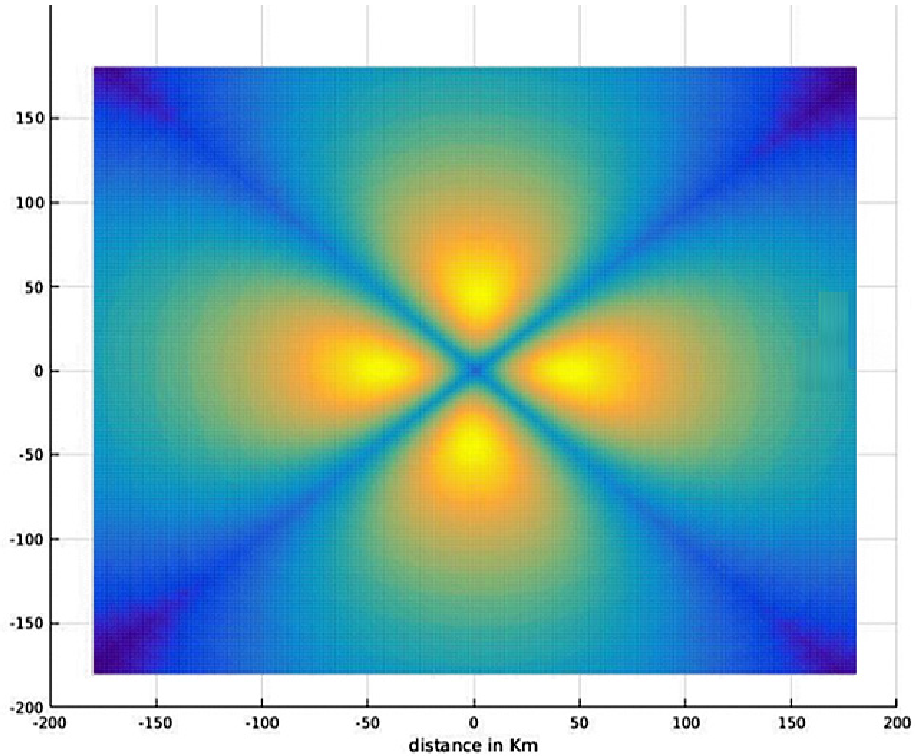


Figure 4-15 ***Dual-Rickenbacker antenna pattern.***
Note low SINR along diagonals caused by sectors interfering with each other.

Electronics

Each of the independent two-sector Rickenbacker systems was a cross between a small cell, which typically has more integrated electronics, and a macro base station architecture, which typically has separate baseband and radio systems.

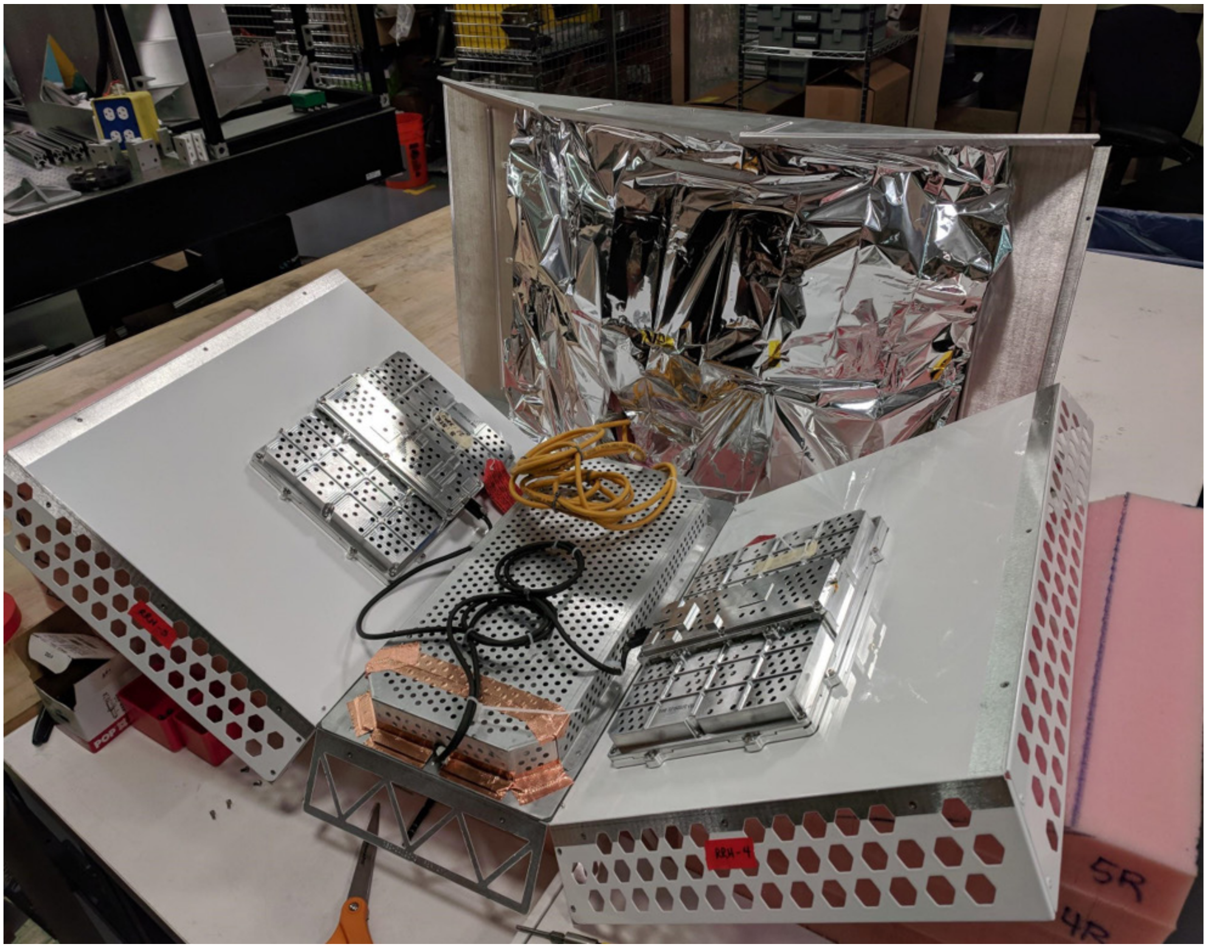


Figure 4-16 ***Rickenbacker electronics.***
Central EMI enclosure contains the Capri baseband board and two-side enclosures mounted behind each antenna; each contain a Soundblox RF system.

Rickenbacker has a single baseband unit (BBU), named Capri, connected via a digital interface (similar to the industry-standard CPRI interface) to two Remote Radio Head (RRH) systems (though they were only inches away). The RRH consisted of a Soundblox Core containing the RFIC transceiver and an RF board (Soundblox Bxx, where xx is the LTE band number). These RF boards were LTE band-specific, so different regions required different boards.

Loon's initial Rickenbackers were Band 20 (Soundblox B20) to support Kenya and other African countries and Band 28 (Soundblox B28) to support South America. These bands are in the 700 MHz to 900 MHz range. The propagation characteristics of this low frequency made it possible to attain a reasonable coverage area with decent signal level using minimum weight and power. Similar coverage areas with higher frequency bands would require more sectors, with correspondingly higher mass and power.

Like the single-sector system, the baseband board was similar to the baseband portion of a commercial Nokia small-cell but was designed by Loon for the stratosphere and to integrate into the overall Rickenbacker architecture.

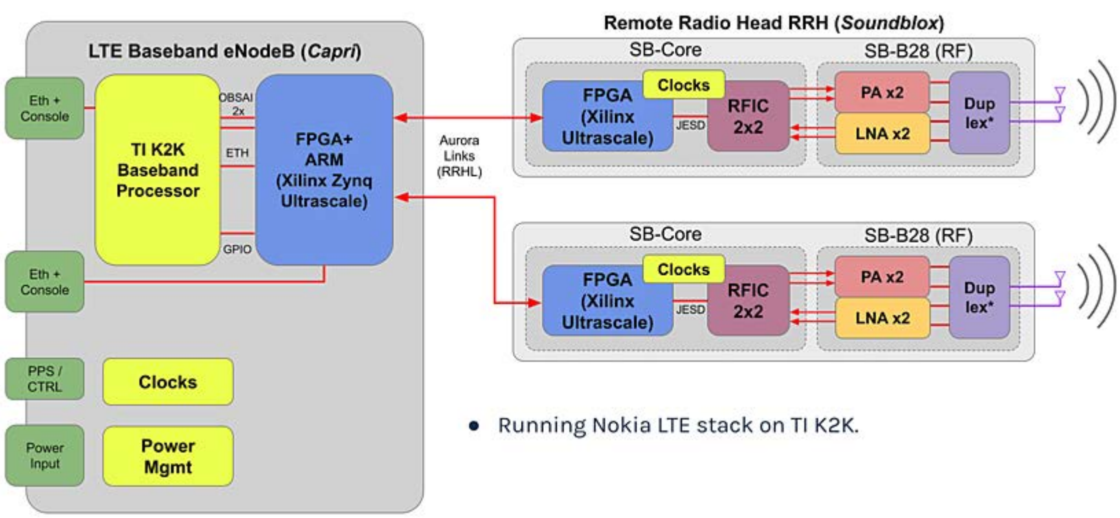


Figure 4-17 Rickenbacker hardware architecture overview.

Several typical base station components rely on near-sea-level air pressure. As you ascend from sea level, the decreased air density causes the insulation provided by the air to drop significantly. For some high-powered electronics, this allows sparking (arcing) at lower voltages across the component, damaging and potentially destroying the part and the board.

(Interestingly, a full vacuum significantly *increases* the insulation, requiring much higher voltage to cross a gap. However, this effect doesn't start appearing until about 65 km (3x Loon's altitude) where the insulation stops decreasing and starts instead to increase, approaching a maximum level as the air pressure approaches zero.)

Loon encountered this issue with the very first LTE systems it flew and quickly changed the filters it used to types that did not rely on air as an insulator.

Mechanical

As with all Loon payload systems, Rickenbacker was designed for minimum weight while retaining sufficient strength to withstand a worst-case burst event and to enable easy recovery after landing. It was divided into two units (rather than a single four-sector unit) in order to fit the four larger antennas onto the existing balloon bus mounting structure. This split architecture compelled the use of two separate baseband units (one for each Rickenbacker), which had a beneficial side effect of increased performance, though with a slight increase in power consumption and mass.

Thermal

Both generations of LTE systems used the Loon photon harvester technology, which performed two key thermal functions:

- It acted as a heat sink, pulling the heat from the electronics and exposing it to the cold ambient air, to cool the electronics during the day
- It absorbed infrared radiation coming up from the earth to reduce the heater power that would be required to keep the electronics from getting too cold at night

Without the harvesting of energy coming up from the surface of the earth at night, a relatively large amount of heater power would be needed to keep the electronics safe from cold thermal damage. For more information, see "Thermal" on page 152.

However, for the Rickenbacker, there was so much power to dissipate during the day that the photon harvester could not keep the electronics warm enough at night. It required a great deal of heater power (half of the operating power consumption on cold nights). This necessitated more batteries and solar panels and decreased the amount of energy for ACS maneuvering and LTE service. Loon was looking to solve this problem using different approaches for future payloads. This is discussed further in "High-Capacity Future Access System" on page 377.

Software

The LTE base station central processor ran Loon's own Loonix operating system, described in "Embedded Software" on page 294. Running on this OS, the Nokia base station software stack made up most of the remainder. Loon also added software to enable the base station to integrate with Loon's network management system described in "Airstream–Loon LTE Network Management" on page 196 and included additional monitoring, control functions, and security features for authentication and encryption.

Future Access System (FAS)

Although the dual-Rickenbacker delivered far more capacity than the Fodera system, Loon knew that an even higher performance LTE system would be needed. The next-generation LTE system would need to reliably reach many more users in cars and buildings and across a wider footprint and increase the revenue each balloon could generate when it was serving. The system was expected to be flying in 2022 and would be designed to support the 5G standard, not just LTE. It was named the Future Access System (FAS).

The other objective of the FAS design was to leverage Nokia's existing designs as much as feasible to reduce the work required by Loon, both initially and for each iteration or advancement by Nokia. The baseband unit would use unmodified or very lightly modified commercial-off-the-shelf Nokia boards from their macro-cell base stations. Loon would design the enclosure and thermal management to accommodate this system in an environment quite different from its original intended environment.

The BBU would communicate with several separate Remote Radio Heads, each of which was a mechanical steering gimbal, on which was mounted a multi-sector reflector antenna. The antenna sizing and sector counts were not finalized though the tentative plan was to have three or four gimbals, each with four or seven sectors. Each of these gimbals could point their beam anywhere from directly down almost up to the horizon and at any azimuth (left/right) direction.

Three or four such gimbals would still only cover a fraction of the potential footprint of a balloon, so an additional set of coverage sectors would overlay the entire footprint with a dual-Rickenbacker-like pattern.

Figure 4-18 shows a coverage pattern for a few future balloons operating over southern Kenya, each with a FAS payload that includes the four-sector coverage pattern plus two capacity gimbals.

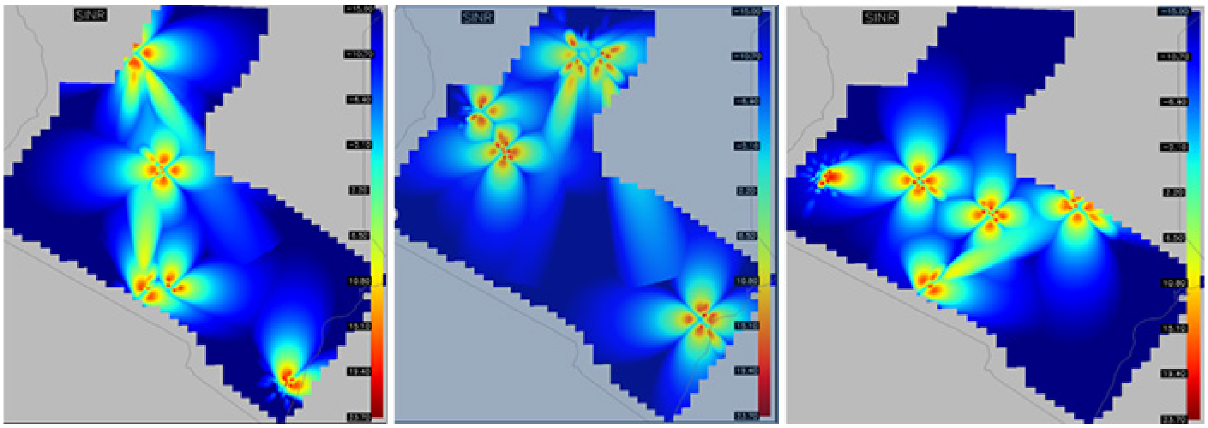


Figure 4-18 ***FAS Coverage pattern.***
Multiple sectors per gimbals allow steerable high-capacity coverage.

More information on the FAS can be found in "High-Capacity Future Access System" on page 377.

Airstream-Loon LTE Network Management

Loon's LTE network is unlike any other in some very key ways:

- Loon's constant motion brought base stations into and out of the network frequently and randomly, requiring live installations and de-installations of dozens of base stations per day.
- When in the network, Loon's sectors underwent dramatic changes in interference impact to each other and the terrestrial network due to their motion relative to each other.
- Similarly, this continued motion meant that Loon's sectors were constantly rearranging such that their "neighbor tables" (list of nearby sectors) needed to be frequently updated.
- The same Loon base stations installed on one operator's network in the morning might fly to and get installed onto another operator's network by the afternoon or the next day.
- Loon base stations will fly right up to and over the partner operator's service region borders and even the country's borders (when both countries had given Loon overflight permission).

- Loon base stations may be operating perfectly but may entirely or partially lose their backhaul connection due to a rain-out condition at a ground station site or other disconnection caused by balloon motion.

Standard LTE networks are quite different:

- Base stations are semi-permanently installed at fixed locations and commissioned once into a network, an involved, largely manual process requiring significant upfront analysis before choosing an exact site, then, after installation, multiple rounds of measuring and reconfiguration.
- When installing a new base station in an area already occupied by base stations, a careful analysis is performed to site, aim, and configure each sector to minimize the interference impact of the new sectors onto the existing network and vice versa. Typically, parameters for the current network need to be modified to better accommodate the new addition.
- Once installed, base stations are left installed and active for years. As a result, it is rare (and very intentionally rare) for a base station's backhaul to disconnect.

Loon's operation also added some very real constraints:

- For example, it was not possible, nor desired, for the partner to:
  » Configure and constantly reconfigure the many base station, EPC, and network management parameters necessary to allow Loon's base stations to integrate or be managed in any way similar to how the partner's base stations were integrated and operated, or,
  » Understand or support the rapid change of interference environment and neighbor relations that Loon's sectors underwent.
- It was impossible to do live measurements of all the base stations from the ground because those base stations were entering and leaving the service area at random places along the perimeter. Additionally, they often moved so fast that any attempt at a typical "drive test" would fail because the balloon and base station being measured might have moved itself hundreds of kilometers during that drive.

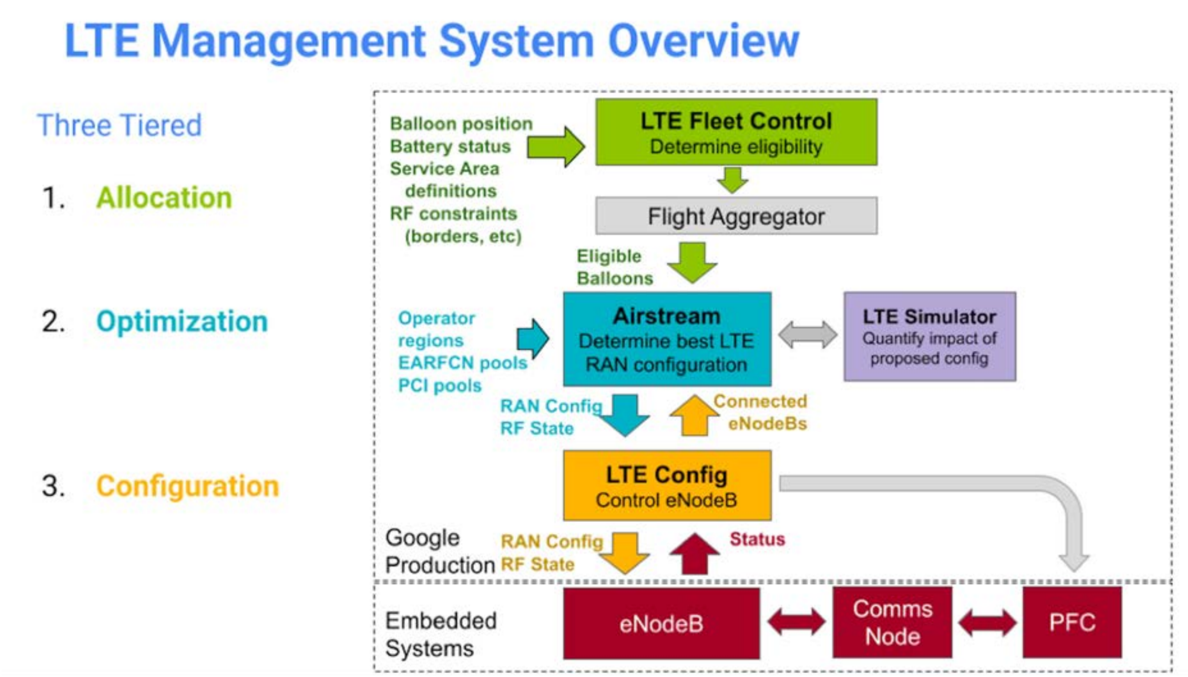


Figure 4-19 ***LTE management system overview.***
Airstream plays key role in optimization of system but it relies on the rest of the network and fleet management systems.

Loon's LTE management system, the core of which was Airstream (Figure 4-20), was created to do this constant configuring and reconfiguring as Loon's base stations come into and wander around a service area and to monitor all systems for anomalies constantly. Some of these functions are similar to the 3GPP's Self Organizing Network[2] (SON) mechanisms, which were defined to allow networks to install and reconfigure networks more quickly, but Airstream needed additional capabilities. Altogether, this system performed the following functions:

- Continuous, periodic (once every 12 minutes) re-optimization of a sector on/off and transmit power level to maximize service objectives such as the number of users and throughput per user while meeting RF power-flux density constraints to minimize interference in-country across country borders.
- These optimizations performed a series of LTE simulations with a variety of different LTE sector configurations to quantify the impact to the important service metrics of each of those configurations, based on detailed interference simulation with Loon sectors and other known co-channel (same channel) sectors.

2 https://www.3gpp.org/technologies/keywords-acronyms/105-son

- Dynamic adjustment of LTE cell selection and attach sensitivity parameters and other controls (e.g., qRxLevMin).
- Accumulation of key LTE performance metrics for later incorporation into control algorithms.
- Live modification of neighbor lists, similar to 3GPP's Automatic Neighbor Relations to enable smooth handovers.
- The complete LTE Management system, including Airstream, could be monitored and controlled from anywhere in the world, no matter where the balloons or partner network were located. Exceptions and alerts were automatically generated when systems exceeded defined bounds and humans were notified in an orderly and well-defined process. As a result, no large permanently staffed sea-of-screens Network Operations Center was required.

Optimization

Control algorithms were developed and tuned to optimize for various parameters:

- Physical Cell Identifiers (PCIs) were assigned using graph vertex coloring, thus ensuring optimal use of PCIs without any duplication
- Sector on/off and channel assignment leveraged a simple greedy algorithm to maximize network coverage
  - » Simple greedy also enforced limitations on how many changes could be made per optimization cycle to minimize disruption for connected users.
  - » Simple greedy worked quite well, but machine learning approaches showed some promise, particularly for higher levels of control (including channel and bandwidth, many more sectors, steering of sector-groups, etc.)
- An in-house algorithm optimized cross border interference to balance PFD constraints with link budget constraints of UEs near the border

Future versions of Airstream would have also included channel and bandwidth selection and several other configuration parameter optimizations.

Core Network

Loon's Core Network system included the following:

- 3GPP-defined[3] LTE EPC[4] components:
  - » Mobility Management Entity (MME)
  - » Serving Gateway (S-GW)
- Diameter Routing Agent (DRA)
- Loon Network Router (LNR), principally for establishing encrypted tunnels to ground stations and acting as a mobility anchor for those tunnels

Loon's core network is unique in its architecture and reflects the challenges specific to a balloon-based solution that augments traditional mobile networks.

Existing mobile networks and the components in those networks, including the core network, are typically designed assuming the base stations are not moving; that only the users are mobile, and the base stations are fixed to a position. As a user moves around, the core network manages the user's mobility as they hand over from cell to cell. In Loon's case, in addition to the users moving, the eNodeB (hosted on the balloon) is also moving, not just around the service area but also completely leaving the service area (and country, and continent) and then returning days or weeks later.

To abstract the operator's core network from Loon's base stations entering and leaving service, Loon's engineers adopted a split EPC design. The design separated the core network into two major sections: the base station and radio-related components (MME, S-GW), which would be in a Loon-managed core, and the remaining subscriber-focused components (P-GW, HSS, etc.), which would remain in the MNO partner's existing core network. Figure 4-21 on page 203 shows the breakout of the eNodeB base stations, Loon Core network, and the MNO's core network, all interconnected using standard 3GPP interfaces.

3 https://www.3gpp.org/technologies/keywords-acronyms/100-the-evolved-packet-core
4 https://en.wikipedia.org/wiki/System_Architecture_Evolution#Evolved_Packet_Core_(EPC)

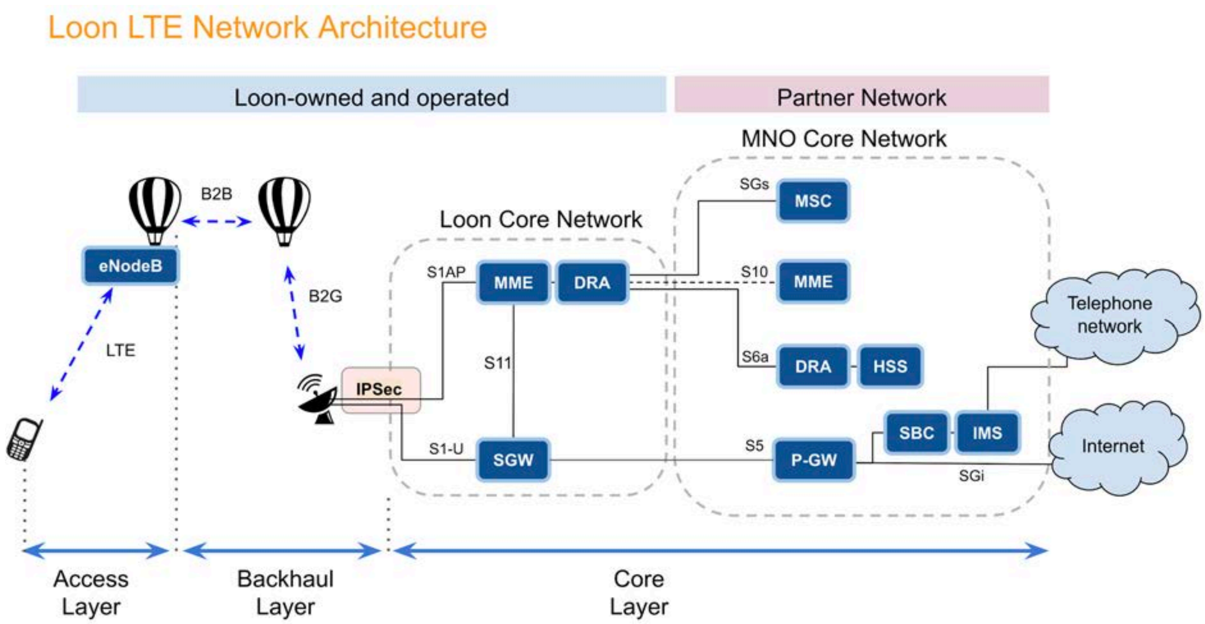


Figure 4-20 *Loon LTE network architecture.*

The separate network functions in a core network (e.g., MME, DRA, SGW, PGW, and HSS) have historically been implemented in "network appliances." Each function is performed by an individual purpose-built hardware box containing functionality-specific electronics to enable the required network function. However, most modern networks are transitioning to a "virtualized" network function architecture in which these appliances are replaced by standard computers (servers). As a result, they have sufficient processing speed and storage to perform all the required network functions in software running on standard processors in those servers. In some cases, hardware-based packet acceleration and handling are included in the server, but such acceleration is fully customizable by software to perform virtually any network function. Loon took a fully virtualized approach from the beginning because of the similarity of this architecture and approach to Google data center software development and deployment.

This approach is called Network Function Virtualization (NFV). The overall architecture was defined by the international standards organization ETSI and describes which blocks perform which functions and how they interface to each other, etc. Loon used a proprietary NFV infrastructure based on internal Google systems and had a mix of proprietary and commercial virtual network functions (VNFs), all implemented in software running on standard servers.

Loon leveraged Google's expertise to design and deploy this solution with the latest software engineering best practices. Given the integration with Google infrastructure, rigorous privacy and security policies were put in place. In addition, Google's site reliability engineering expertise helped ensure that Loon could leverage a Continuous Integration/Test/Deployment (CI/CT/CD) pipeline with effective alerting and dashboarding. This allowed for incremental updates and ensured the network could be managed remotely without a traditional Network Operations Center (NOC).

As Loon looked to scale the number of MNO partners supported, the engineering team created multi-tenancy capabilities such that the same core network could support multiple MNOs. Multi-tenancy tends to be very complex given the range of customization and MNO-specific integrations that need to be supported. Loon also built out playbooks to onboard new MNOs and perform the interoperability tests to ensure seamless operation.

The Loon network integrated with the partner's such that a customer on the ground experienced service as they would normally, with no difference in operator indications on their phone and no changes in their billing. Generally, this required Loon to pass through operations such as identification/billing, authorization/policy control, and authentication/security. However, because the partner MNO was paying Loon based on number of GBs served and several other factors, it was necessary to have a deeper, non-pass-through integration between Loon and the partner's billing systems.

Given the timeline and complexity of integrating Loon's network with new MNO partners, Loon looked to roaming agreements and interconnections to significantly reduce this effort and duration. This was also vital for rapidly deploying service when responding to a natural disaster (e.g., Hurricane Maria in Puerto Rico). With this approach, Loon could potentially integrate with a single MNO that has extensive roaming agreements well in advance of a potential disaster. Loon could then use that MNO's existing interconnections to many other MNOs worldwide to enable those affected by a disaster to take advantage of Loon as soon as the disaster knocked out their networks.

Loon did such integration with AT&T's core network in 2018 and 2019 to support a combined disaster recovery service for the Caribbean. It enabled Loon to start providing service in disaster zones worldwide with a much shorter network integration process. However, it was still necessary for ground stations to be installed in or near the affected region, so Loon began to deploy this infrastructure throughout the Caribbean in case it was needed.

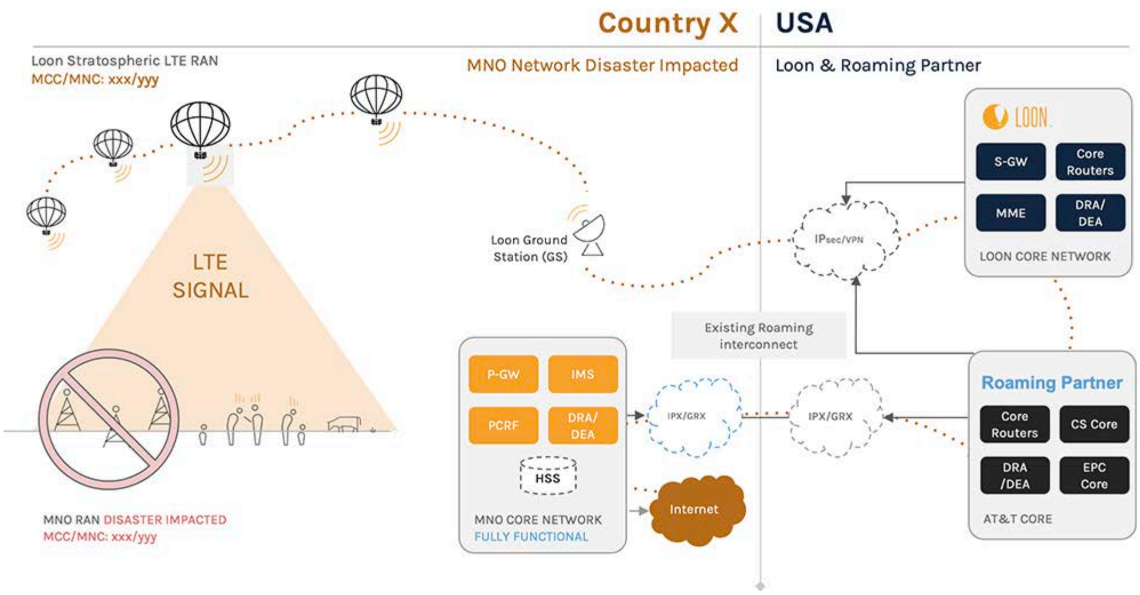


Figure 4-21 ***Roaming Partner Interconnection.***
Rather than interconnect directly with every operator, Loon established a relationship and interconnect with a Roaming partner that had many existing roaming relationships with operators around the world.

Backhaul Network Layer

The Loon backhaul carries the user data and control data between the Loon Core Network and the LTE base station in the vehicles' service payloads, and consists of

- Terrestrial portions: the ground stations and the fiber or other links connecting them back to the core network.
- Airborne portions: the B2X radios and gimbals and the Comms Node and the LAN interconnecting them.

The primary difference between the Loon backhaul and a typical mobile backhaul is that Loon moves and undergoes near-continuous evolution of its network topology, with many links forming and breaking. To isolate the LTE elements (base station and core network), from this behavior, Loon establishes an end-to-end tunnel across the backhaul network as shown in Figure 4-22 and manages the routing of these tunnels as the topology changes.

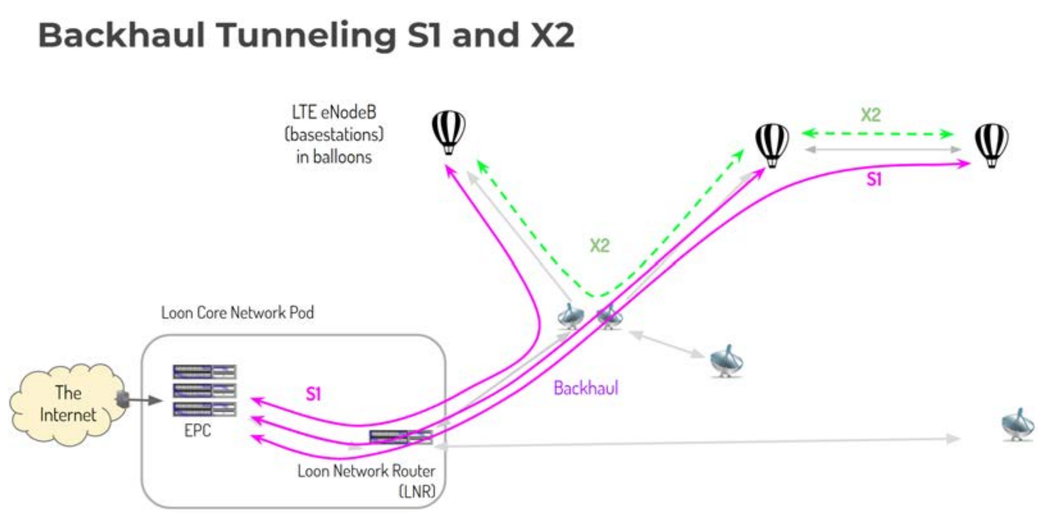


Figure 4-22 End-to-end encrypted tunnels.

Such changes will happen as the balloons move around relative to one another, but they might also happen because of a link or other failure. Figure 4-23 and Figure 4-24 show what happens if two ground stations get rained out. Because B2B links and different B2G links already exist, the network can tolerate this weather event without problem, as shown in Figure 4-24 and Figure 4-25 on page 206.

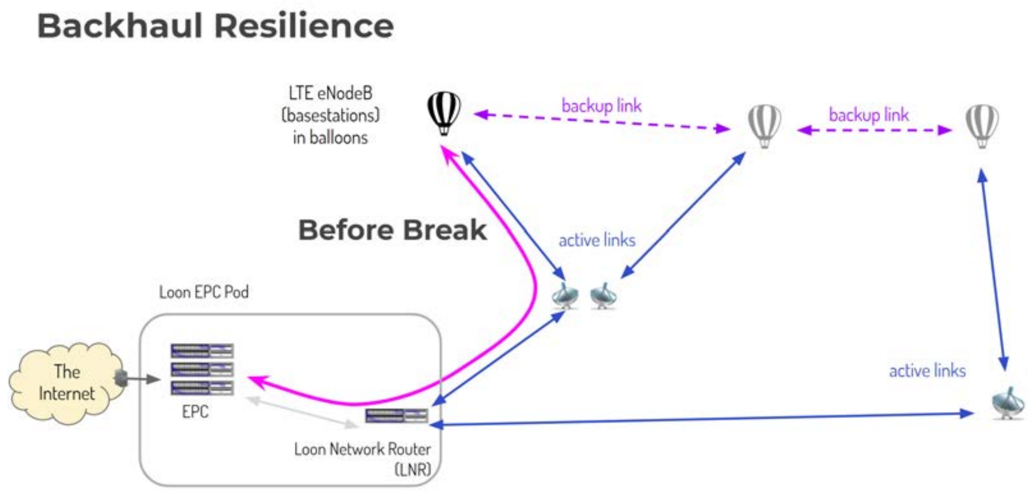


Figure 4-23 ***Backhaul resiliency from redundant mesh (1).***

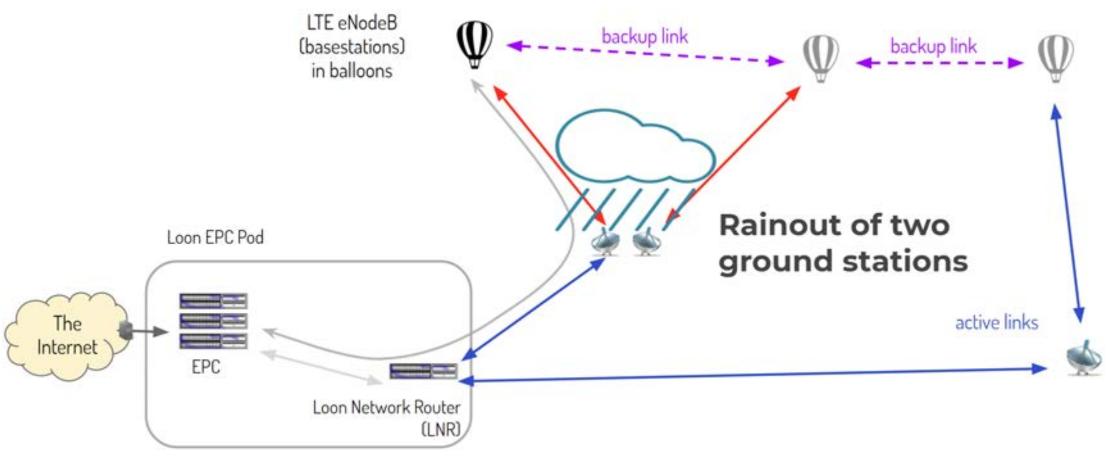


Figure 4-24 ***Backhaul resiliency from redundant mesh (2).***

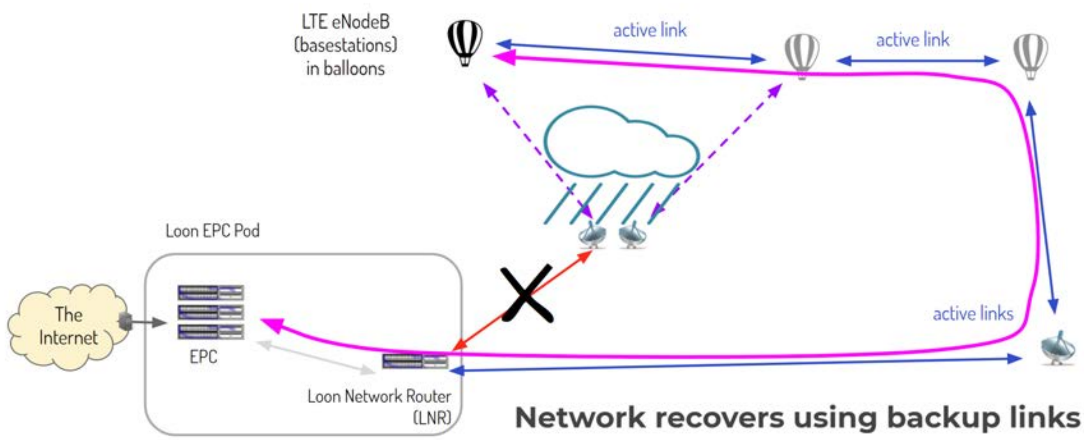


Figure 4-25 Backhaul resiliency from redundant mesh (3).

Layers of network encapsulation in a modern mobile network make it appear to applications running on the users' equipment that they are connected directly to the internet. Under the hood, the user's data goes through a series of encapsulations and de-encapsulations (wrapping and unwrapping) as it traverses the partner's or other backhaul provider's networks. In order to isolate the 3GPP nodes (eNodeB, MME, S-GW, etc.) from Loon's backhaul topology changes, Loon added layers on top of the existing 3GPP-defined networking layers. Loon's TS-SDN orchestrated the creation, modification, and security of these tunnels as the underlying network nodes changed to adapt to balloons entering or exiting the service region.

One of those layering and encapsulation configurations (for the LTE user's data plane) is shown in Figure 4-26.

For a more detailed (and zoomable)view see this artifact: LnArt- Loon Network Data-Plane & SDN Control-Plane.

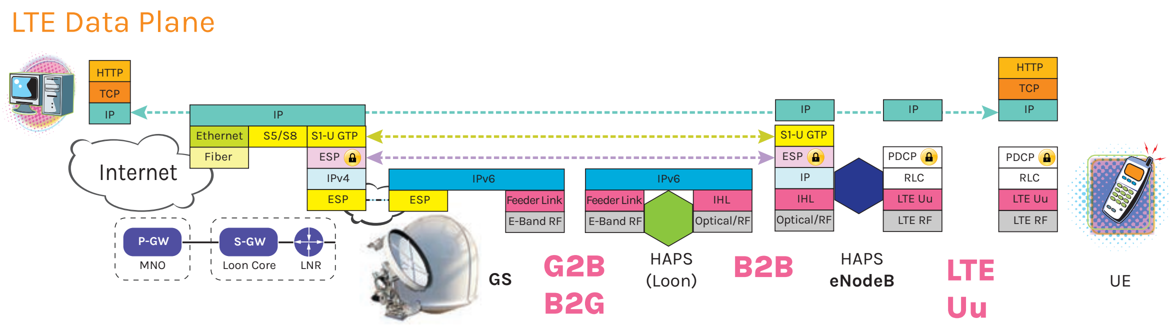


Figure 4-26 Loon network layering and flow diagram–LTE data plane.

Backhaul Network Orchestration - Minkowski TS-SDN

Loon's backhaul network was significantly more difficult to manage than it might first appear:

- As described earlier, Loon's links and nodes are constantly moving.
- These radio links were also impacted by weather, terrain, and distance, and consequently had RF characteristics that were changing constantly.
- In addition, there were many varying obstructions both on the balloons and on the ground that needed to be considered when determining new links but also when existing links will drop.
- The extreme pointing requirements of a fraction of a degree was difficult to achieve on a moving platform that was subject to turbulence and could impact the time to establish a reliable link.
- Relying on satellite communications with very low data rate, very high latency and occasional dropouts, meant that redundancy and resiliency needed to be added to the controlling systems as well.
- Turning this into a high-availability, self-forming and healing mesh in the stratosphere spanning thousands of miles was a significant challenge.

Figure 4-27 ***Mesh of 33 Loon balloons.***
Balloons spanned 3,500 km over Kenya and the Indian Ocean in October 2020.

Traditional Software Defined Networks (SDN) orchestrate network configurations and traffic flows but are incapable of handling nodes moving around or wireless propagation challenges. Therefore, Loon had to develop a first-of-its-kind Temporo-spatial SDN (TS-SDN) to blend the best of Google's SDN technologies with a physics engine capable of modeling motion and RF propagation. This system was code-named Minkowski, after the mathematician Hermann Minkowski[5] who, according to Wikipedia, showed that Einstein's special theory of relativity could be understood geometrically as a theory of four-dimensional space–time.

5 https://en.wikipedia.org/wiki/Hermann_Minkowski

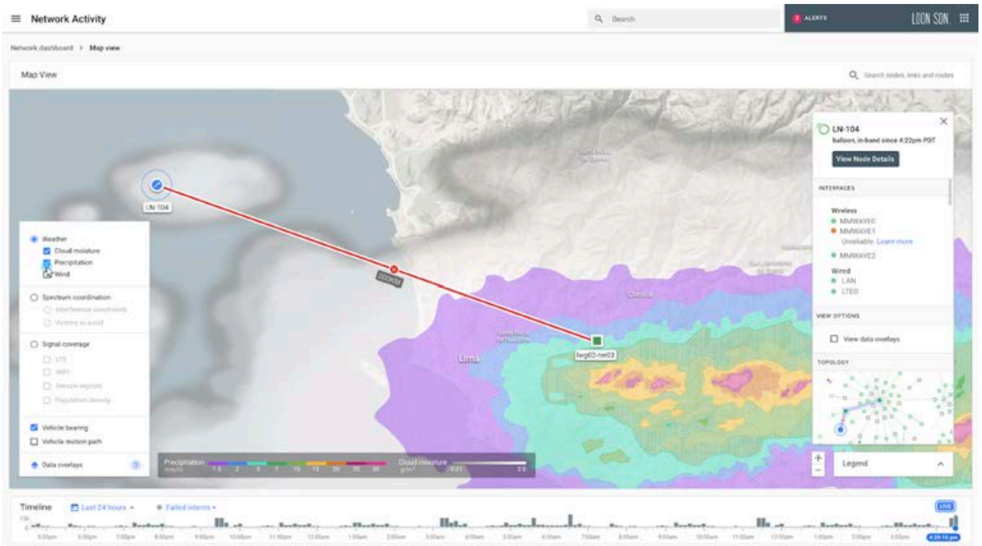


Figure 4-28 ***Loon's Network Operation Weather UI.***
Poor weather impacting a wireless link.

The Minkowski TS-SDN co-optimizes wireless link planning, radio resource management, and routing across terrestrial and non-terrestrial network segments based on end-to-end connectivity requirements. It uses predicted motion and weather forecasts to avoid service disruption by anticipating when existing links will diminish or drop and when others will become available.

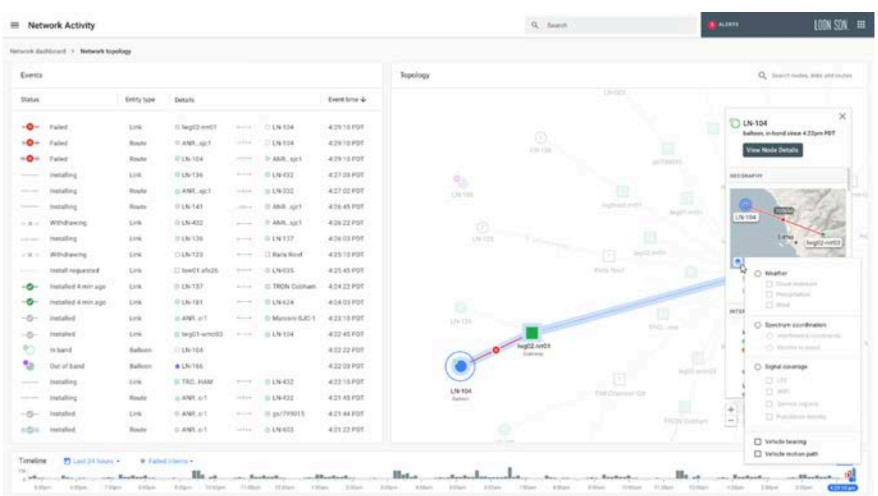


Figure 4-29 ***Loon's Network Operation UI Overlays.***
Motion, weather, and RF overlays for network debugging.

Though required for Loon, Minkowski was designed to be general-purpose and applicable to many other types of networks with time- and/or spatially-varying topologies, particularly those supporting multiple different RF bands and optical links. Key capabilities include:

- Designed for networks with highly directional links with mobility along and above the surface of the earth:
  - » Models transmitter and receiver antenna patterns
  - » Computes link vectors for all objects
  - » Propagates link budgets forward in time
  - » Tracks data rates and propagation delays
- Supports RF links: ITU radio frequency propagation models:
  - » Computes attenuation due to free space, gaseous absorption, clouds, rain, and fog; accounts for Doppler shift and speed of light delay
  - » Validated, industry-standard models implemented at scale
  - » Interface with regulatory spectrum-licensing databases for interference avoidance
- Supports Free Space Optical Communications (FSOC) links:
  - » Computes FSOC propagation through vacuum and atmosphere
  - » Models validated with Loon and third-party-measured data from the troposphere and stratosphere
- Live and forecasted weather data for all link computations:
  - » Utilizes data from 140,000+ streaming weather stations
  - » State-of-the-art forecasts from NOAA and ECMWF
  - » Ability to fuse data from Loon or other mobile or stationary weather stations
- Scalable compute with cloud-native code:
  - » Utilizes best-in-class Google compute resources
  - » Automatically scales to consume additional compute cores

- Intent-based software-defined networking:
    - » Pluggable solvers describe the desired state of the network
    - » The network controller continually monitors and updates network devices to align with the intended state
- Intuitive web-based Network Operations Console for easy debug:
    - » Provides historical analysis of network states and transitions
    - » Visualization of network routes and link heuristics
    - » Map and globe views to visualize weather, terrain, trajectory, and other network metrics

Loon's TS-SDN built on existing SDN best practices and extended the functionality and APIs to incorporate its time and spatial nature and support for heterogeneous wireless links. The control to data plane (southbound) interface (CDPI), for example, was extended beyond the standard OpenFlow-based interface to support SDN control of steerable beam tasking, radio resource management, and schedulable time of enactment. In addition, the system was designed for delay and disruption tolerance leveraging scheduled and sequenced control messages and concurrent control channels (in-band and out-of-band).

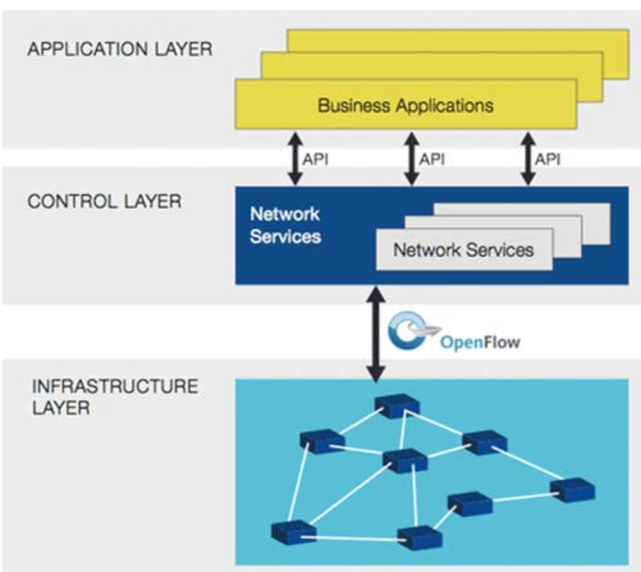


Figure 4-30 Standard SDN architecture.

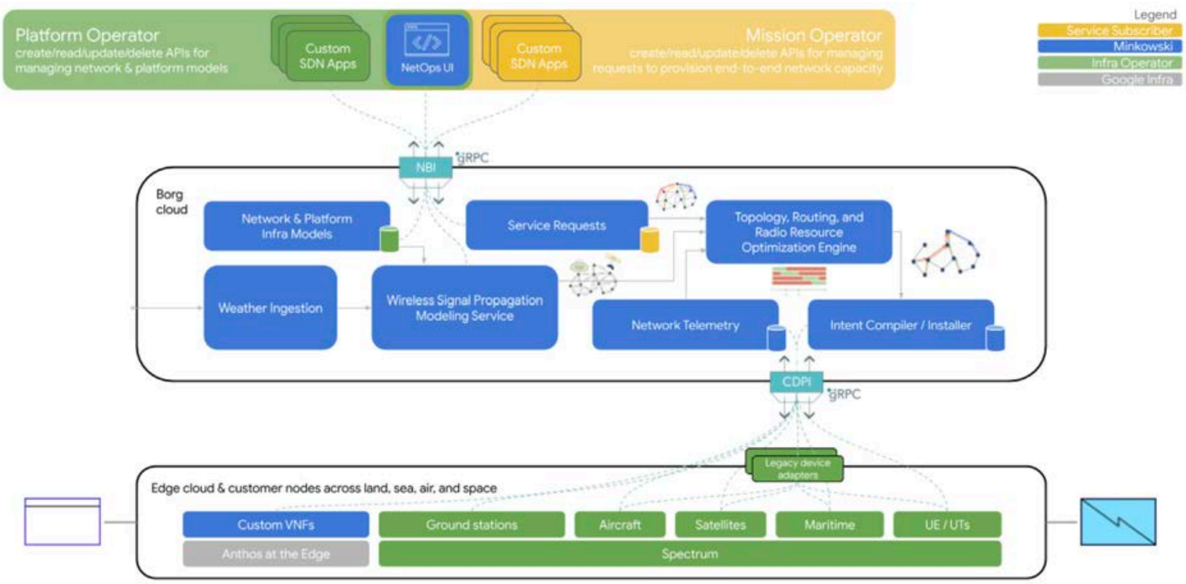


Figure 4-31 ***Loon's TS-SDN architecture.***

The control layer runs on Google infrastructure and performs the following key functions

- Ingest weather data from ECMWF and local weather sources (e.g., from Loon-installed weather stations at ground station sites)
- Ingest telemetry from network nodes (ground stations, balloons, etc.)
- Aggregate service requests from the network operator (e.g., Loon)
- Store and process system definitions:
  - » Static:
    - – Node network information (e.g., interfaces and MAC address)
    - – Node physical information (e.g., shape of the object, position, orientation, fields of regard, and obstructions)
    - – Node RF information (e.g., RF bands, bandwidths, power, antenna patterns, and interference constraints)
    - – Visualization models (for the Network Operations UI)

  - » Dynamic:
    - – Transceiver link reports including changing link budget, bitrate, and propagation delay due to link motion
    - – Drain provisions
    - – Route provisions
    - – Node trajectory

- Solver engine- the solver has been tuned and adapted to optimize for network availability, and efficiency through several systems and software approaches:
  - » Intent-based network specification
  - » Demand-aware topology formation
  - » Multipath routing to ensure network resilience
  - » Built-in hysteresis to reduce network churn
  - » Link learning- adapting to actual-world performance of links
  - » Incremental solving by building on a previous result for faster control
  - » Opportunistic make before break
  - » Compatible with Mixed Integer Programming (MIP) based solver
  - » Compatible with machine learning-based solver

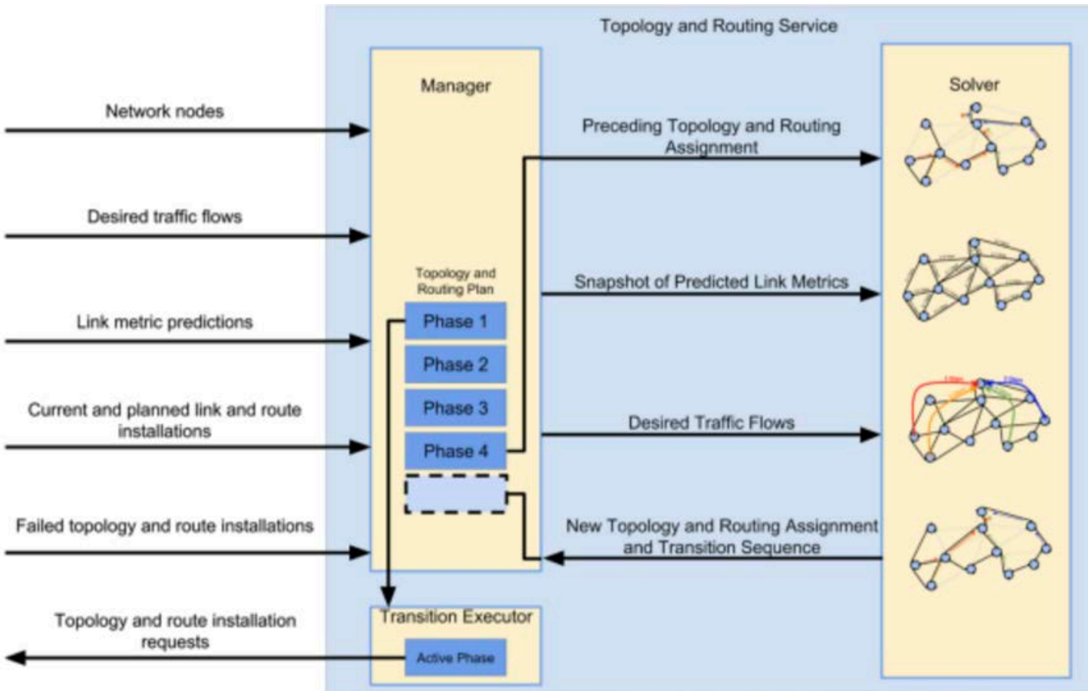


Figure 4-32 Control layer's topology and routing logic.

Loon's TS-SDN benefitted from extensive simulation testing and real-world analysis. As the team tackled new corner cases and deployment realities, new tools and techniques were implemented:

- SDN probes to periodically ensure all nodes in the system are healthy
- Post-deployment ground station obstruction-mask corrections. As the world changes around a ground station, a new obstruction may render the original obstruction mask obsolete (e.g., a new building or billboard). The team used a camera mounted on the ground station to rebuild the obstruction mask.
- Leverage of ad-hoc networking protocols like BATMAN to enable resilient routing across the nodes in the mesh
- Advanced network debug interface allowing the operations and development teams to visually understand the sequence of events and the changes to nodes/links across time and space.
- Scaling link budget calculations by using S2 cell binning.
- Seamless key rotation. Ensuring that the backhaul is not disrupted while rekeying IPSec tunnels to ensure that data transiting the network is secured.

Loon's TS-SDN was designed to support use cases beyond Loon's balloon network. The system's capabilities lend themselves to orchestrating Low Earth Orbit Constellations (LEO), such as Telesat's solution featuring multiple satellites, ground stations, and large numbers of user terminals.

Unlike the Loon use case, which relied on the LTE subsystem to manage the access layer, Minkowski enhancements were planned to manage the access layer as well. This would include full Airstream functionality but also more advanced beam hopping and shaping. Incorporation into Minkowski would also allow a combined access and backhaul optimization and orchestration for applications where that access-aware backhaul optimization (or vice versa) leads to better overall system performance.

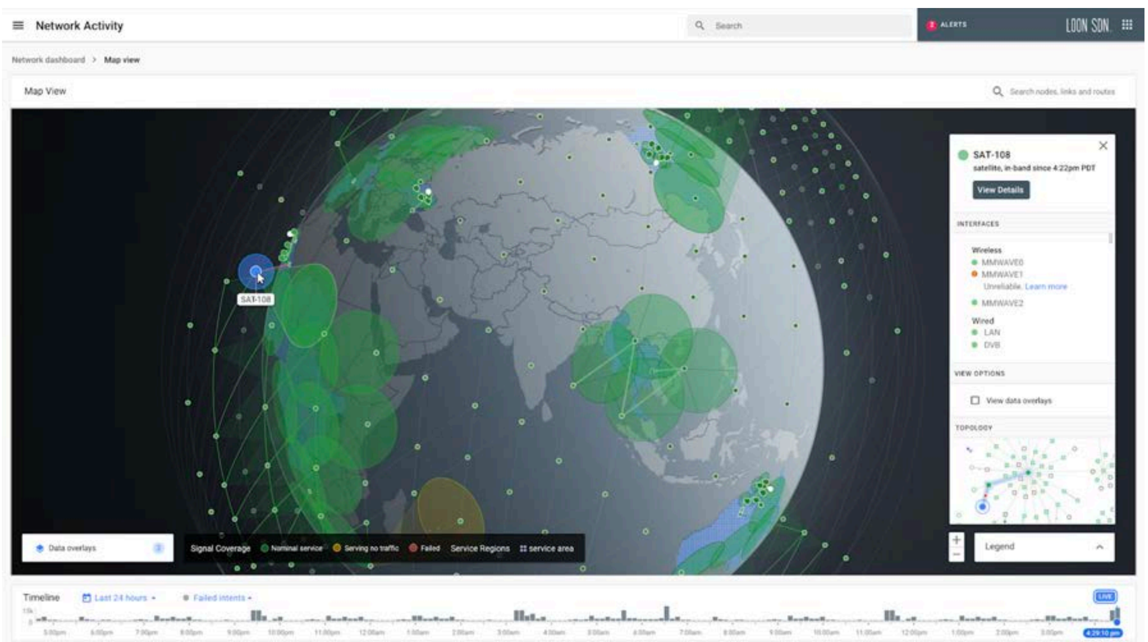


Figure 4-33 ***TS-SDN network operations UI showing the LEO use case.***

Loon envisioned a multi-network future where the TS-SDN would seamlessly orchestrate terrestrial (4G/5G), high altitude platform station (HAPS), and satellite networks (LEO/MEO/GEO) based on the user's connectivity requirements and the global optimization of resources.

For additional information on how the system operated and lessons learned, please see the white paper *Loon: Designing & Operating an Aerospace Mesh Network* on the Loon website.

Ground Station

The Loon ground station system provides a gateway link to Loon's stratospheric network of balloons through a high bandwidth point-to-point wireless connection. The deployed system utilizes the E-Band spectrum with FDD channels from 71 to 76 GHz (uplink) paired with 81 to 86 GHz (downlink) occupying 750 MHz of channel bandwidth in each direction. The ground station can route network traffic (control and user data) between the balloon and Loon's Core Network system.

The ground stations are deployed in a geographically diverse manner around or adjacent to desired service areas. In addition, multiple ground stations are deployed at each site to provide site-specific redundancy, increase B2G connections to the fleet, and offer redundant full 360° azimuth accessibility. The combination of geographic diversity and site-specific redundancy creates a highly available ground network approaching 99.9% availability.

In addition, regional ISPs or telecom operators typically provide ground station connectivity to the internet through their fiber networks. The Loon backhaul system is controlled by the Loon TS-SDN controller, which orchestrates ground to balloon and balloon to balloon links and optimizes routes across the Loon network.

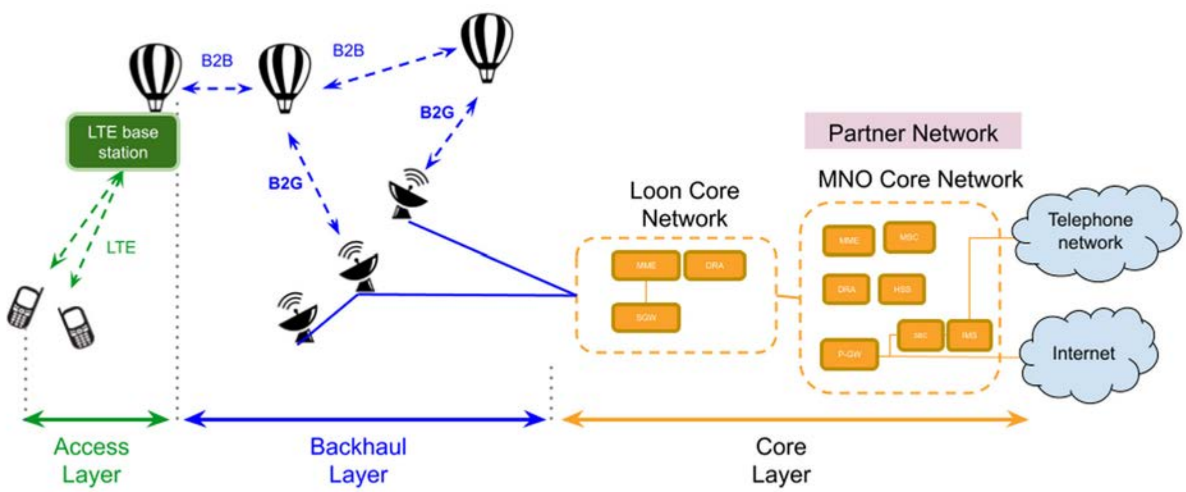


Figure 4-34 Loon backhaul network topology.

Commercial Marine Satellite Terminal

Loon worked with a commercial provider, Cobham (SeaTel), to develop a customized solution to support tracking Loon's stratospheric balloons from fixed terrestrial and mobile shipboard deployments. High-rate tracking functionality was added through software upgrades to support HAPS drone vehicle connections (see "HAPSMobile Sunglider–Rickenbacker LTE Gen-2, B2X and CommsNode" on page 232). The system supports three-axis stabilized operation with unlimited Azimuth tracking between elevations of -10° and +95°.

The system is environmentally ruggedized to support operation from -25° to +55°C with a weatherproof enclosure. The system is designed to be mounted on rooftops, telecom towers, and ships. A rapid deployment system has also been developed utilizing a 20-foot Conex box to transport two ground stations and associated supporting equipment and then serving as a mounting platform for the ground stations. The control and data interface is provided through two strands of single-mode fiber with redundant bi-directional interfaces. The system is powered through telecom standard -48VDC with a typical power consumption of 110W.

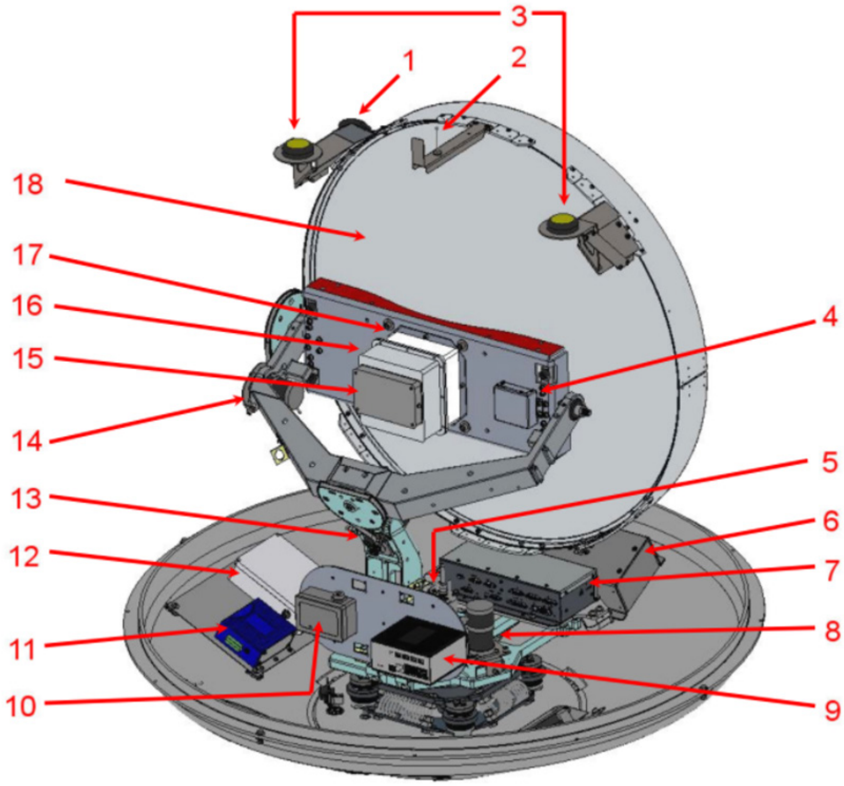
Figure 4-35 Loon Ground Station (left) and Radome (right) and two ground stations mounted on rapid deployment Conex.

Antenna

The antenna is a three-foot Cassegrain parabolic reflector custom designed to support E-Band and to be integrated into the Marine Satellite Terminal. The antenna provides 53 dBi of gain with a 0.3-degree beamwidth across the E-Band spectrum and utilizes a single (horizontal) linear polarization. The combination of narrow beamwidth and horizontal polarization limits the potential for interference with existing fixed terrestrial systems, which predominantly use vertical polarization.

Loon Adaptation and Interfacing

Loon worked closely with the commercial partner Cobham to adapt the system to meet Loon's unique requirements. Loon provided a custom E-Band radio design and additional commercial OTS equipment to support system control, balloon and HAPS tracking, and system calibration. Figure 4-34 shows the Loon-provided components.



1	System GPS Antenna	10	ADS-B Receiver
2	ADS-B Antenna	11	Ethernet Media Converter
3	GPS Antennas (2)	12	Controller Module
4	Motion Platform	13	Cross-Level Motor and Belt
5	Fiber Rotary Joint	14	Elevation Motor and Belt
6	DC-DC Converter	15	Dual-Antenna GPS Electronics
7	Tracking Control Unit (TICU)	16	Radio System, E-Band
8	Azimuth Motor and Belt	17	Feed Assembly
9	Managed Gb-Ethernet Switch	18	Antenna Reflector Assembly

Figure 4-36 ***Loon Ground Station Key Components.***
Loon-supplied components are indicated in blue.

Loon Modules

- Loon's controller module (12) provides the interface to the ground station through an electrical GbE port. All user data traffic is encrypted via an IPSec tunnel between the ground station and Loon's EPC. SSH access provides the interface for engineers to remotely service and debug the ground station.
- The Ethernet media converter (11) supports an SFP-based fiber interface converted to electrical to interface with the controller module.
- The Managed GbE Switch (9) is configured by the controller employing VLANs to support traffic flows for data, control, and management functions.
- The ADS-B Receiver (10) and antenna (2) allow for tracking of the balloon position that the controller utilizes to command the COTS Tracking Control Unit (7) to point the ground station antenna for link acquisition.
- The Radio System (16) provides real-time RSSI voltages to the COTS Tracking Control Unit (7) to support searching functions for link acquisition and maintenance.
- The Dual-antenna GPS Receiver (15) and GPS antennas (3) are used for heading calibration purposes should there be changes in position after installation.

The Loon-adapted ground station system provided the means to connect to stratospheric balloons rapidly. Initial targeting for the ground station was accomplished through balloon position information relayed by the ADS-B receiver. Search functions controlled by the COTS controller were then utilized for link acquisition. Position information coupled with direct RSSI feedback to the COTS controller allowed link acquisition and maintenance. In addition, calculations of expected RSL levels based on link distance were constantly evaluated to optimize link performance.

Key Learnings

Calibration of the ground station pointing was critical throughout the ground station life cycle.

Factory calibration was accomplished through tracking RSSI performance while targeting the Sun. The E-Band receiver provides 5-6 dB of signal when targeted directly at the Sun on a clear day. Analyzing the tracking data over a four-to-five-hour period provided key data to determine cross-level calibration trim levels for the COTS controller.

Pre-installation calibration was utilized at the deployment site to validate that the calibration values had not changed during the shipping and delivery process. Calibration verification was also accomplished through Sun tracking.

Post-installation calibration was utilized as a final verification of the installed system through Sun tracking.

Post-deployment balloon tracking verified initial calibration and data on the specific Azimuth and elevation error level.

Radome

The radome for the system is designed to support low insertion loss in the E-Band spectrum and serves as an environmental seal for the ground station system. Target insertion loss of less than 1 dB is designed to limit the impact on the link margin for the system. The radome material is ABS plastic with a UV coating formed from four separate pieces, bonded together during the manufacturing process.

E-Band Radio Overview

Loon's E-Band radio system was designed to support both the ground station and the B2X payload system. The radio supported high TX power coupled with low noise RX to provide maximum system performance. It can be configured through software to operate TX in either the 71-76 GHz or 81-86 GHz bands and RX in the alternate band. This was required for the stratospheric mesh where every B2X must support connection with every other B2X. This high/low configuration is not required for the ground station radio. The because the ground-to-balloon (TX) link is always set to the 81-86 GHz band to help avoid interference with radio astronomy bands. Several channels within each of the bands were available to reduce interference between nearby B2X links.

Radio Architecture

The E-Band radio front end utilizes direct conversion mixers to convert baseband differential IQ input to E-Band frequencies for the transmit function. The receiver front end uses an LNA, direct conversion mixer, and baseband amplifiers to create a differential IQ baseband output. Circulators are utilized to allow either TX or RX to be selected for each band (71-76 GHz and 81-86 GHz). Baseband switches are used to select the band of operation.

The E-Band front end interfaces with the baseband processor, which contains the modem controller, modem, MCU, GbE interface, and associated power and control functions. The ASIC modem and controller contain embedded software to configure and control the modem. These parts require a license to operate but are generally available. The modem modulates signals from the GbE PHY, demodulates signals received from the front end, and delivers them to the GbE PHY. The current Loon design supports a single GbE interface for data, but the modem is designed to support a 10G interface.

Radio Electronics Hardware

The hardware design consists of multiple PCBAs mechanically packaged into a custom aluminum enclosure. The system supports a fan module for cooling and all necessary interfacing connectors to operate within the ground station system. The E-Band frontend is a custom Loon design that combines chip and wire technology and SMT. The custom design limits the choices for contract manufacturing partners to those who support chip & wire capability. Due to the complicated design, Loon could only achieve ~85% first pass yields of the frontend module, which is a very high failure rate. Rework of the chip and wired components responsible for this fallout failed to solve the problems in most cases.

To remedy these issues, Loon was engaged with a highly experienced millimeter-wave transceiver ODM partner to provide a custom design where the partner would build and deliver tested screened modules increasing the yield to 100%. Initial prototype modules were received but not evaluated.

The baseband processor PCBA is a conventional SMT design that most contract manufacturers can manufacture. The PCBA contains the ASIC modem, modem controller, MCU, and GbE PHY. In addition, the board provides high-speed connector interfaces to the frontend module and a simple interface board supporting GbE to the ground station.

Key Learning

The box-level build of the radio system requires extensive RF capability and experience manufacturing millimeter wave products. Equipment and fixturing supporting E-Band radio testing require specific RF skill sets and extensive capital investment. The mechanical enclosures needed to support the E-Band radio design contain critical radio antenna elements to the antenna waveguide interface. Therefore, supplier selection for the radio mechanics should be limited to suppliers with extensive experience manufacturing millimeter-wave mechanical enclosures.

Radio-Embedded Software

The radio system contains embedded software to control the E-Band frontend and modem interface. Key features controlling the radio include band selection, ACMB, LDPC, FEC, muting, etc. In addition, the embedded radio software provides an API to the ground station controller to support the management of all radio functions.

Wi-Fi B2G as Test and as Backup

Loon also designed an 802.11 Wi-Fi-based (balloon) radio and ground station as a precursor to the E-Band system. The Wi-Fi ground station was deployed for testing and validating the comms systems on the balloon platform. The Wi-Fi ground station is based on a similar Commercial Marine Satellite Terminal as the E-band ground station. The WiFi system cannot support balloon-to-balloon links. The result is a 1:1 relationship between ground stations and balloons. In addition, European regulations for the outdoor use of Wi-Fi spectrum specifically limit EIRP to a level that would not support ground to balloon links. Therefore, while this system was initially successful in supporting disasters in Peru and Puerto Rico, it is not practical for commercial application.

Site Support Cabinet

There is a single Site Support Cabinet for up to four ground stations. The cabinet provides the ground station system interfaces for data, power, and management functions. It can also serve as the demarcation point between the ISP/Carrier network and the ground station. Out-of-band management is also supported by the cabinet through a connection to the ISP/Carrier network that is on a separate path from the in-band network. Most Loon installations utilized an outdoor cabinet with environmental controls to support a ground station site with up to four ground stations. Backup batteries provided up to eight hours of power to help ground station operation during power outages as long as the ISP/Carrier network connections were active. A serial console server in the cabinet provided OOB troubleshooting capability for the ground station system.

Packet Handling, Routing, and Security

The ISP/Carrier router typically provided a fiber connection for each ground station. The demarcation point was determined on a site-by-site basis depending on the ISP/ Carrier router location. Routing requirements are outlined below:

- Must be able to accept /64 IPv6 and /28 IPv4 globally routable prefix lengths
- ISP must advertise Loon IPv4 and IPv6 default routes only using BGP
- It is not desired for ISP to announce full IPv4/v6 Internet routing tables
- Support BGP session authentication using one of either RFC2385 or RFC5925 with key
- Support eBGP multipath with ECMP traffic balancing for situations where more than one Loon to ISP interconnect link is used
- Support Generalized TTL Security Mechanism (RFC 5082)
- Support additional BGP Operations and Security Best Current Practices (RFC 7454)

Figure 4-37 ***Installation of radome on Loon Ground station.***

B2X Subsystem

The Service Payload included three B2X systems. Each consisted of an E-band radio and a 43 dBi parabolic reflector antenna mounted on a controllable gimbal that could rotate in two dimensions: in Azimuth (side-to-side) and Elevation (up-and-down), to be able to point to a ground station or another balloon.

Each of the axes of motion is controlled by a direct-drive brushless DC motor and an optical encoder.

The main electronics boards are housed on the outboard side of the elevation motor opposite the 43 dBi E-band dish antenna.

The Azimuth (left/right) stage provides 350° of range, and the Elevation stage (up/down) supports 242° range, centered around nadir, or "straight down." These motions enable the antenna to be pointed anywhere from directly below the balloon up to 31° above the horizon and across an azimuth range of a full 360°.

Loon investigated other options besides a two-axis (2D) gimbal, including a 2D phased-array or a 1D phased-array + 1D mechanical. The phased array solutions typically cost more and consume more power for lower-frequency systems but potentially improve reliability and accuracy by eliminating the mechanical axis. However, using E-band, the power and cost premium for a phased array versus a mechanical approach is even higher, making it a clear decision to use mechanical steering.

Due to the tiny beam size of the E-band system, calibration of the B2X and the ground stations was critical. In addition, the drift of the beam angle relative to the inertia measurement units (IMUs) made it necessary to occasionally re-determine the orientation of the gimbal. Therefore, Loon incorporated an accurate, end-of-line calibration step for all devices and performed in-field, live calibration of the B2X using received signal strength and link success tracking to monitor misalignment and drive the alignment process.

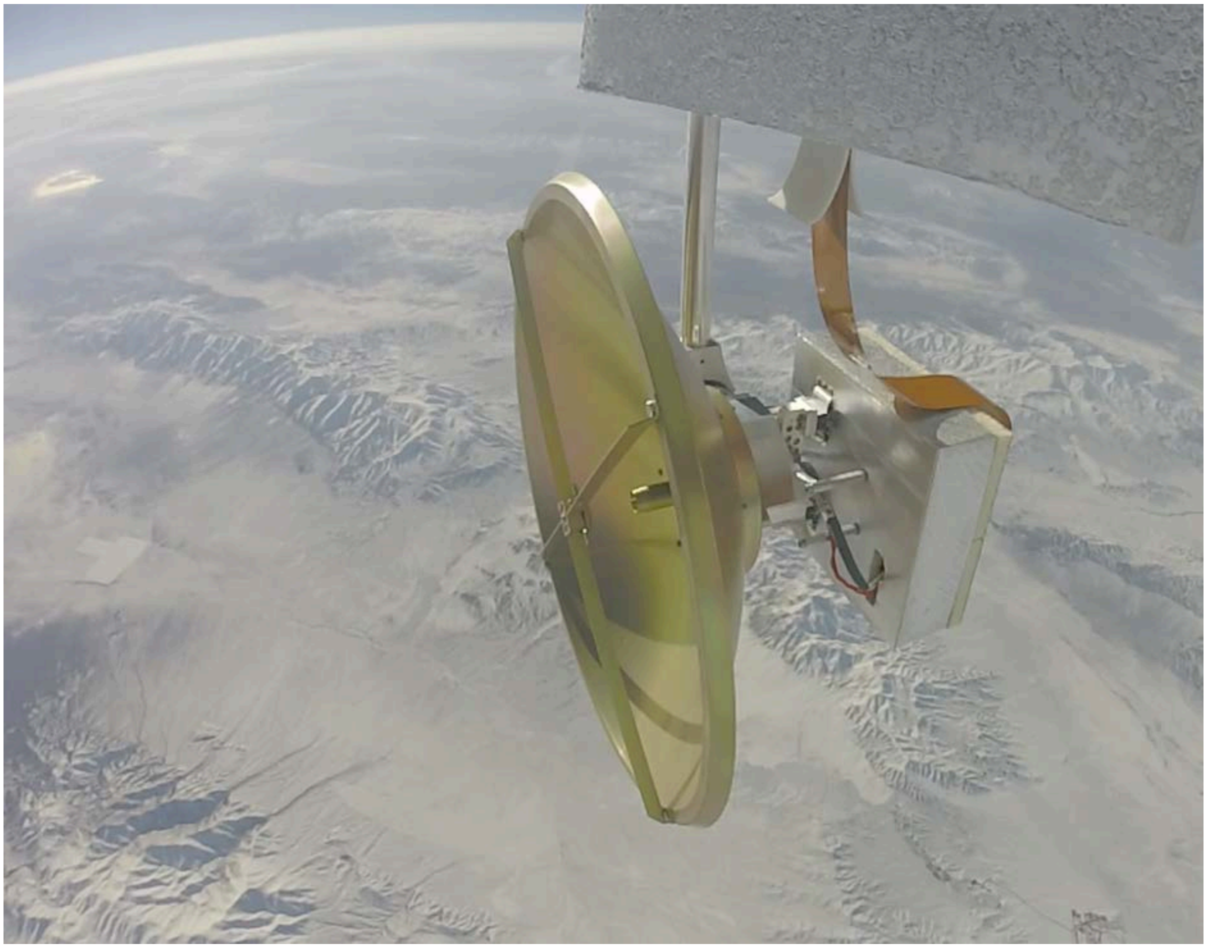


Figure 4-38 B2X radio system.

Architecture and Electronics

Several microprocessors on the boards provided all of the functionality to control the motors and pointing as well as the higher-level operations of the modem control. ran various software, including systems running both Both of Loon's embedded operating systems: Major Tom (FreeRTOS-based) and Loonix (ChromeOS-based) were used on the B2X system.

- The *Morello* Motor Module controlled the two motors, AZ and EL, which stand for Azimuth (left-to-right) and Elevation (up-and-down), to point the gimbal anywhere.

- The *Newbeck* (and earlier, *Brubeck*) Modem + Controller Module contains the actual modem, which deals with digital and analog baseband signals sent to and received from the Radio Frequency (RF) Module as well as the gimbal pointing control, which interfaces with the Motor Module.
- The *Amelia* RF Module housed the radio analog front-end electronics. The Power Amplifier (PA) for transmission and the Low-Noise Amplifier (LNA) for reception as well as the high/low band switching to enable any B2X to connect with any other B2X was the output that directed a high-powered signal outward to the other B2X.

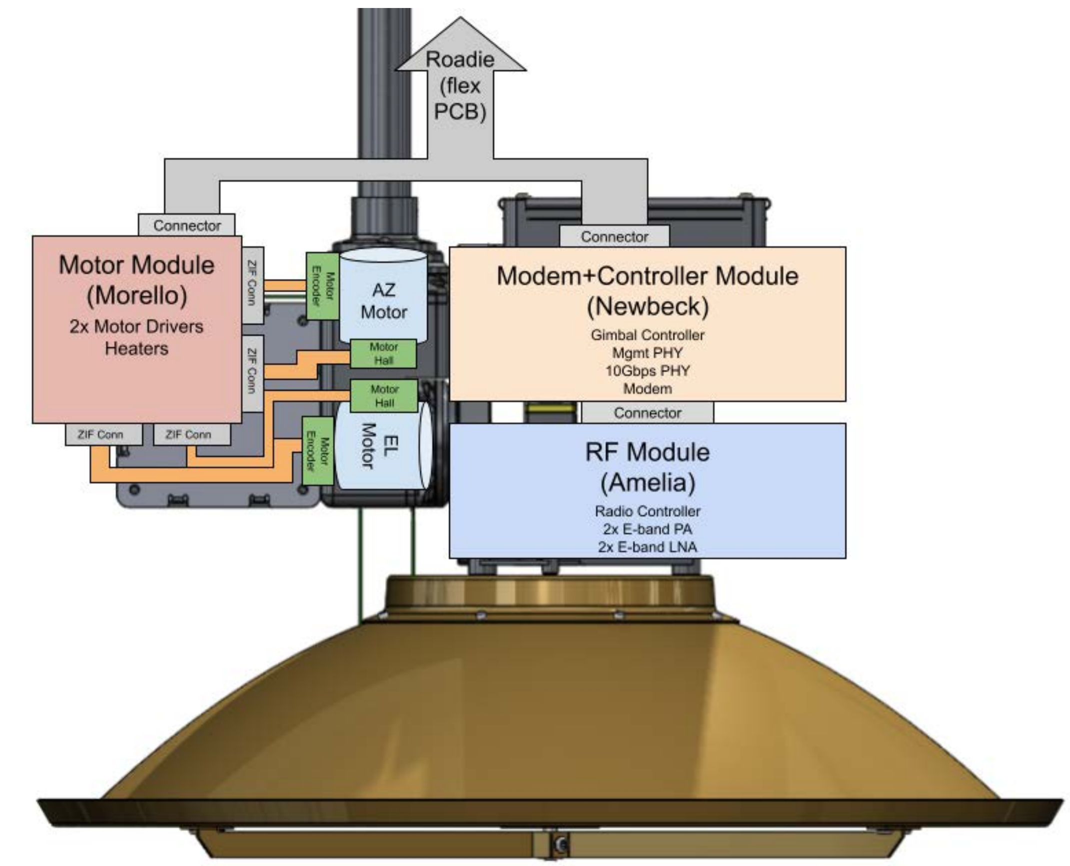


Figure 4-39 B2X printed circuit boards and flex.

Mechanical

- Though Loon engineers were initially skeptical, Flex printed circuit material (FPC) proved sufficient to last 200 days in the stratosphere, across many flex cycles, and in a wide variety of temperatures.

Calibration

The B2X subsystem was designed to support both open-loop (no feedback) and closed-loop (with feedback) pointing. In this way, the significant majority of pointing actions would occur open-loop, making pointing faster because there would be no delay caused by an additional beam search step. To enable reliable open-loop pointing, and to ensure that closed-loop results could be used to calibrate the system in-flight, attention was paid to the potential areas for miscalibration and methods of recalibrating on the fly.

The mechanical system and potential areas of misalignment can be represented as follows:

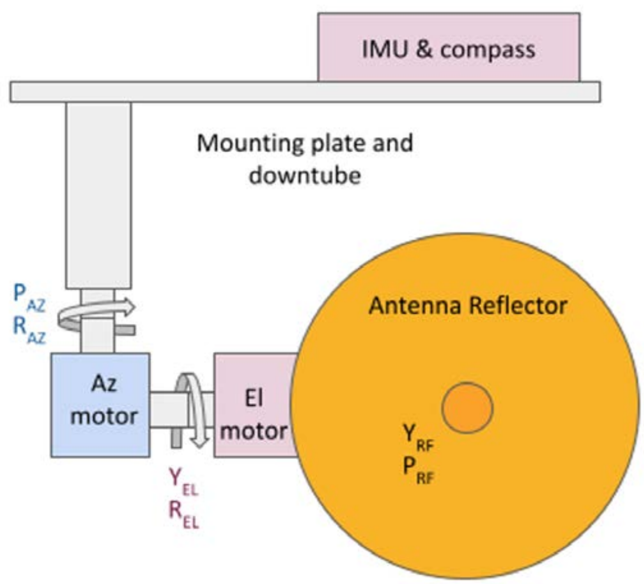


Figure 4-40 Calibration axes of B2X gimbal.

All sources of angular misalignment between the IMU/compass and the RF beam are of concern, but the following potential areas of misalignment appeared to contribute most of the error:

- P_{AZ}: Pitch misalignment between azimuth motor and IMU/compass, primarily due to non-orthogonality in mounting plate and downtube.
- R_{AZ}: Roll misalignment between azimuth motor and IMU/compass, primarily due to non-orthogonality in mounting plate and downtube.
- Y_{EL}: Yaw misalignment between elevation motor and IMU/compass (taking into account offsets in azimuth motor rotor position due to azimuth motor encoder offsets).
- R_{EL}: Roll misalignment between the elevation motor and azimuth motor due to non-orthogonality in the machining of the interface between these two motors.
- Y_{RF}: Yaw misalignment between RF beam and elevation motor due to manufacturing tolerance of reflector and reflector mount.
- P_{RF}: Pitch misalignment between RF beam and azimuth motor, due to (a) manufacturing tolerance of reflector and reflector mount, and (b) offsets in the elevation motor encoder.

We use a two-step process to calibrate these error angles:

Step 1 (pre-flight):

- The gimballed antenna is mounted in a precision-machined frame.
- The antenna is commanded to point to a nominal position: 0° azimuth, 0° elevation.
- Three laser rangefinders are mounted in a plane with exactly 0° pitch and 0° yaw. The laser rangefinders measure the distance between this plane and 3 points along the rim of the reflector antenna. Simple trigonometry is then used to compute the actual pitch and yaw angle between this known plane and the plane of the reflector rim.
- The result of this calculation is programmed as the PRF and YEL variables, while all others are set to nominal 0.
- Note that the pre-flight test does not measure the direction of the actual RF beam; instead, the plane of the reflector's rim is used as a proxy.

Step 2 (during flight):

- Each time an RF link is formed between a payload antenna and another balloon or ground station, the following data is recorded:
  - » Latitude, longitude, and altitude of this antenna
  - » Latitude, longitude, and altitude of target
  - » Attitude (orientation) of this payload (from IMU and compass)
  - » Antenna azimuth and elevation delta/error required to close the link (following a successful search pattern)
  - » Any existing calibration values (typically just P_{RF} and Y_{EL})
- After a few days of in-flight operation and data collection, a set of linear equations can be solved to determine the resulting Azimuth and elevation deltas in terms of payload position and attitude, target position and existing calibration values (the knowns) and gimballed antenna misalignment angles (the unknowns).
- Due to the large number of measurements, this becomes an over-determined set of equations, which the software used the least-squares algorithm to solve. This gives each of the six misalignment angles, and the negation of each becomes the correction angles.
- These correction angles are programmed into the gimballed antenna motor controller during flight.

Pre-flight calibration (step 1) resulted in an average of 0.7° open-loop accuracy in the final azimuth/elevation pointing direction of the RF beam. However, after in-flight calibration, we typically obtained at least 0.2° accuracy.

Service Monitoring Probes

Loon developed many service monitoring tools but two of the most important were focused on determining the availability and quality of Loon service at a specific spot on the ground:

- Rømer monitoring box
- Virtual Probes

The remote monitoring box was named Rømer, after Ole Rømer[6], a Danish astronomer who, according to Wikipedia, made the first quantitative measurements of the speed of light. The Rømer was a standalone service monitoring system, capable of continual periodic measurement of the Loon service. It consisted of a Google phone, for connecting to and testing the Loon LTE network, and a Raspberry Pi, for managing the tests as well as connecting to a local ISP or SATCOM provider for communication with Loon's cloud systems. Originally, a Nexus 6 was used but this was later changed to a Google Pixel or Pixel 2.

Software on the Raspberry Pi together with software on the phone, would periodically wake up the phone, attempt to connect to the Loon LTE network and then store the results. The results would occasionally be queried by the Loon LTE monitoring system, accessing the Rømer boxes through its local ISP connection.

The advantage of this device was that Loon or the partner MNO could deploy the device anywhere of interest and it would continuously monitor the availability and quality of the Loon service at that location, 24 hours a day, 365 days a year.

In addition to the physical Rømer device, Loon developed a purely software-based service monitor, called a Virtual Probe. Unlike the Rømer, which required a logistically complicated deployment in the service area, Virtual Probes could be deployed nearly instantly across thousands of different locations. They would perform a nearly identical function to the Rømer boxes: continually monitor the Loon service at a specific point.

The Virtual Probes would track all relevent vehicle and service-related telemetry and indicate that the location was served at a particular sample time if the following was true at that time:

- A simulation of all active Loon LTE sectors in the area show that the probe's location has sufficient SINR for a phone to successfully attach and transfer uplink and downlink information to a Loon sector.
- That Loon sector's health monitoring metrics indicate that it is functioning correctly at that time.

These conditions will only be met if:

- One or more balloons have an active LTE sector where that sector's antenna pattern covers the tested location.
- Those balloons have a functioning backhaul connection to a Loon core network.

6 https://en.wikipedia.org/wiki/Ole_R%C3%B8mer

- That core network is functioning and is connected to the MNO's core network and all interconnections and interfaces appear to be functioning.

Through the use of these tools, Loon actively tracked availability and other service metrics across all of our active service areas and used the measurements to generate service reports for our MNO partners.

Loon Communications Payload Offering

Loon's communication payload was versatile and adaptable to similar applications with constraints on weight and power usage. In addition, the hardware and software systems packaged together could be leveraged as a solution for other HAPS operators to enable connectivity on their flight vehicles.

HAPSMobile Sunglider–Rickenbacker LTE Gen-2, B2X and CommsNode

Loon and HAPSMobile collaborated to demonstrate LTE from the stratosphere using HAPSMobile's Sunglider, a solar-powered airplane. Loon provided the Rickenbacker two-sector LTE band 28 system and a single B2X unit, together with additional components needed to integrate with the Sunglider platform, including the Loon CommsNode.

Figure 4-41 ***Loon and HAPSMobile.***
Loon team members finalizing integration of Loon payload into HAPSMobile Sunglider aircraft in late 2019.

HAPSMobile announced this collaboration and development in April 2019. The successful demonstration concluded in September 2020 and was announced in October. See the following for links to a video of the flight and a live video conference carried through the Loon B2X and LTE payloads from the stratosphere:

- HAPSMobile and Loon First in the World to Deliver LTE Connectivity from a Fixed-Wing Autonomous Aircraft in the Stratosphere[7]
- Alphabet and SoftBank's solar-powered drone provides first LTE connection[8]

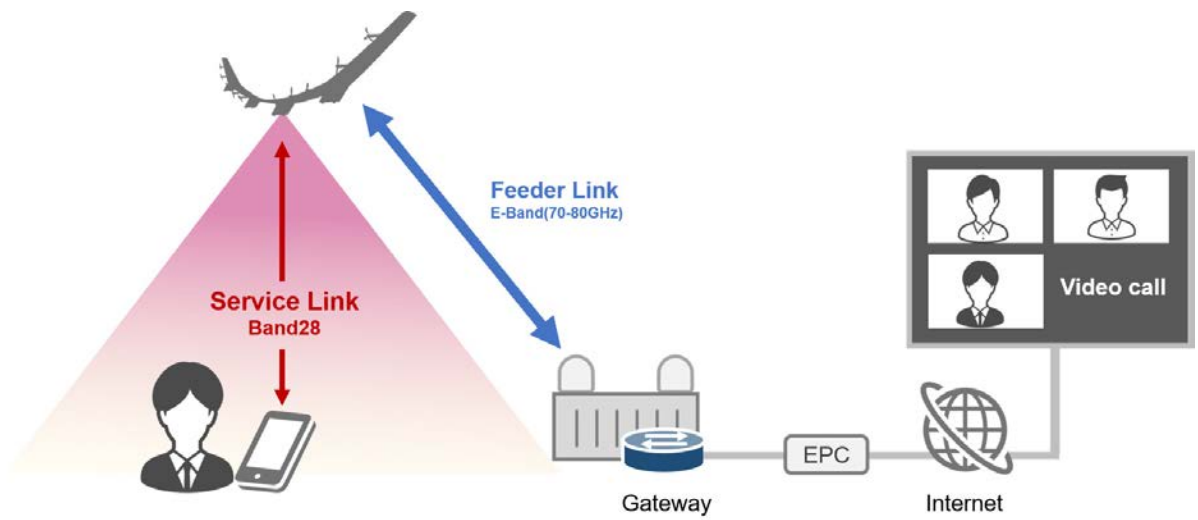


Figure 4-42 ***Loon service payload, ground station and network management software equips HAPSMobile Sunglider to support LTE video call.***

---

7 https://www.hapsmobile.com/en/news/press/2020/20201008_02/

8 https://www.theverge.com/2020/10/8/21507397/alphabet-softabank-solar-power-autonomous-drone-lte-connection

Chapter 5

Loon Software

This chapter discusses the software and applications developed by Loon. This software is found across the system: embedded on the balloon itself, ground stations, and the Loon Core Network, or running in the cloud.

Loon Software Overview

Loon software was distributed across multiple computing environments, which included:

- ***Cloud.*** The majority of Loon software was running in Google data centers and fell into three categories:
  - » Fleet Management and Automation. Described in this section.
  - » Simulation and Analysis. Described in this section.
  - » Network Orchestration and Management. Described in "Communications Architecture and Evolution" on page 172.
- ***Other data centers.*** Loon's core network software and several other functions were running in remote data centers, either in, or near our partner operators' central offices or own data centers. See "Core Network" on page 200.

- ***Embedded on the vehicle.*** Approximately 30 different computers on the vehicle, across the balloon, bus, and service payload, ran everything from real-time avionics software to LTE base station stacks. The software running these systems is described in "Embedded Software" on page 294. The avionics are described in "Avionics, Bus and Power Systems" on page 96 and the B2X and LTE systems are discussed in "Service Payload Architecture" on page 178.
- ***Embedded on the ground station.*** Ground station control and management software and other backhaul-related functions ran on embedded systems within the ground station or at the ground station site. This software is described in "Ground Station" on page 216.
- ***Laboratories, Testing, and Production facilities.*** Loon developed many engineering tools, production, and testing equipment for use in Loon facilities or those of its supplier partners. Although these tools were essential to Loon's success in developing, manufacturing, and operating its service, they are not discussed in this document.

Fleet Management Systems

Loon's fleet management software, running in Google data centers, was not a single monolithic system but was distributed across many separate binaries and services, essentially a microservices architecture.

Need for Full Automation

In all cases, Loon's objective was for maximum safety in all Loon operations, which mandated reliable automation of the many complex hardware and software systems on the vehicle and of the complex software systems in the cloud that were used to navigate and orchestrate the vehicles. To ensure the automation and management systems were functioning, Loon also built extensive monitoring and alert systems.

Flight engineers were employed to not just watch and look for anomalies, but had many additional responsibilities, including supervising fleet operations, working closely with Loon launch and recovery operations teams and with domestic and international air traffic control. Some of the software systems supporting these roles are described in this section.

Fleet Management Functions

The primary functions of the Fleet Management Systems were:

- Control and Telemetry sending/receiving, recording and processing: securely, safely and reliably communicating with the balloons
- Vehicle systems and health management: basic controlling and monitoring of the balloons
- Power management: tracking, forecasting, controlling and prioritizing power use, and optimizing solar power harvesting and battery storage
- Aviation data management: allowing automation and humans to understand relevant borders and constraints regarding any airspace with Loon-relevant attributes
- Wind and weather data ingestion, management and monitoring, including storm handling
- Fleet navigation and orchestration: fly the balloons to where we need them and orchestrate the fleet-level behavior
- Network management systems integration: management and automation of LTE and backhaul networks
- Web UI (Smallworld): allows Flight Engineers and other Loon personnel to supervise the fleet and control the automation systems.
- Additional web UIs for other targeted users: allows Air Traffic Control and others to track our balloons and status
- Integration / interfacing with Simulation infrastructure: forecasts active fleet motion and health, and used for systems engineering and business analyses shows a partial, simplified view of Loon’s fleet management systems.

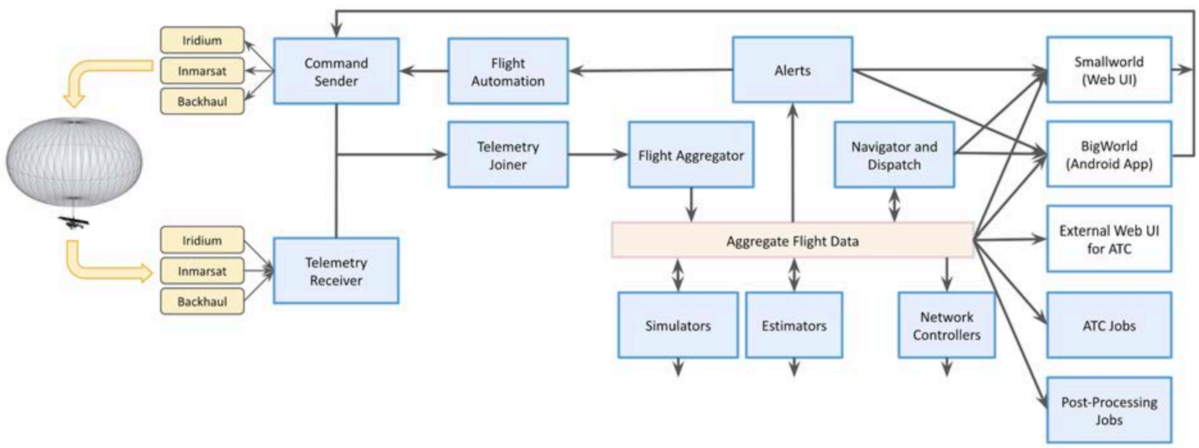


Figure 5-1 *Partial Fleet Management Systems block diagram.*

Control and Telemetry Communications Management and Data Handling

The command and telemetry gateway manages all balloon- and bus-related command and telemetry communications to and from each balloon over low-speed satellite communications and high-speed backhaul links. As described in "Satellite Communication" on page 121, Loon used Iridium and Inmarsat to carry command and telemetry to and from the balloons and also used our own E-band B2X mesh network to carry high-rate telemetry for balloons that had active connections to a B2X mesh.

The gateway directs the commands to the appropriate service based on recent health status and other aspects of the systems. All telemetry from the three channels and all commands sent are joined into a single stream for presentation to the rest of the systems.

Vehicle Systems and Health Management

The Fleet Management software works closely with the onboard avionics to manage all vehicle systems. Functions include:

- Monitoring the onboard power management and distribution systems
- Supplying routine automation and assistance to:
  - » Notify flight engineers or others of certain important balloon or fleet states or required actions, such as need to contact ATC
  - » Perform standard automation sequences such as those during launch and landing
- Tracking the power consumption on all systems
- Forecasting power use for the remainder of the day and the night
- Planning an optimal use of the energy expected to be available
- Prioritizing safety-critical systems and budgeting the remaining power among the steering systems (ACS and lateral propulsion, if present), and communications (service) payload
- Balloon-health monitoring and forecasting, including the health of the ACS, envelope and ballonet, and the volume of lift gas.
- Monitoring for:
  - » Anomalous or otherwise concerning events or parameters across all of the vehicle's software, electronics, and mechanical systems
  - » Tracking against historical data to highlight variances
  - » Generating alerts and triggering automated scripts

Power Management

Loon's vehicle is powered by the sun and is consequently power- and energy-constrained. Power-related software functions are principally oriented around the hardware characteristics:

- From a hardware perspective, the prioritization of power is:
  - » Critical: Core avionics, including SatCom, flight termination systems and any heater power required for these systems
  - » High: Heater power for all other electronics (to avoid damage)

    - High: Navigation for no-fly avoidance
    - Low: Comms and Navigation for service
- It is critical that the balloon never completely runs out of power, which would power-off the SatCom systems (along with everything else), preventing Loon from controlling or monitoring the balloon.
- The minimum hardware that must remain powered are core avionics, SatCom and the flight termination system. The power draw for this must be supported throughout the night and into the morning until the solar panels begin generating power again. Running only these critical elements, and nothing else (such as the ACS or LTE) was called "hotel mode" but was often discussed with and without heater power for the LTE and B2X electronics.
- The largest power consumers are also lower priority vs. the critical avionics load:
    - Comms, consisting of LTE and B2X
    - Navigation, consisting of the ACS pump and, for v1.6 and later, the Seahorse propeller

Forecasting Power Use and Availability

The amount of solar harvest throughout the day is very predictable (in contrast to terrestrial solar) but it does depend on the latitude and season of the year, which affects both the sun angle and, more importantly, the number of daytime versus nighttime hours.

The power consumption of all the other elements of the system, however, is difficult to predict and largely depends on environmental factors. Winds can affect power use, either directly (higher ACS activity due to winds) or indirectly (balloons clumped so LTE is turned off). Air temperature and upwelling IR affects how much heater power is needed (to prevent damage to electronics).

To ensure the system never runs out of power, the power management system places a dynamic minimum state-of-charge constraint, as shown in Figure 5-2. In this diagram, three battery state-of-charge curves are shown, which drive several power management controls:

- ***Optimal*** assumes min constant load, which exactly reaches zero energy reserve at sunrise (for reference only)
- ***Proposed cut-off*** enabled greater comms and navigation power availability but with minimal risk to Critical power consumers
- ***Aggressive cut-off*** assumes nighttime shut-down of comms and navigation enabling maximum power use in daytime

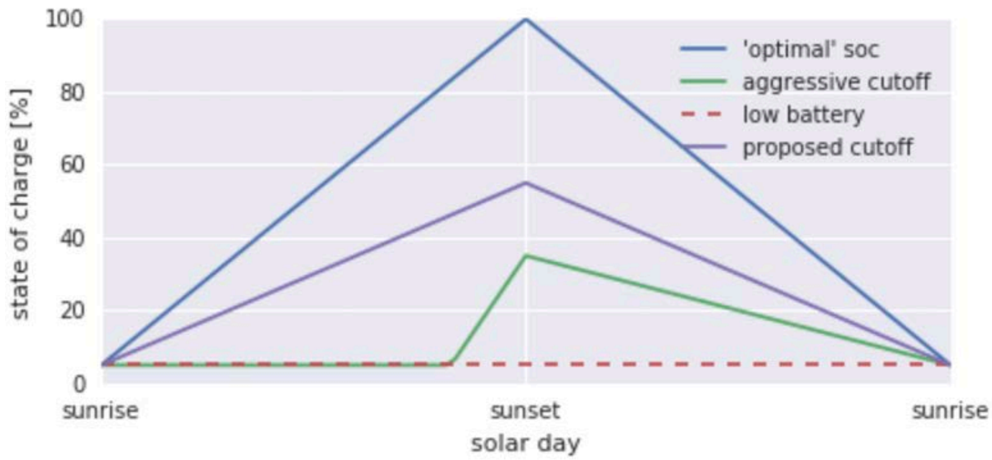


Figure 5-2 ***Battery state-of-charge power cut-off curves.***
From highest (optimal) with minimum consumption limitations, to lowest, with aggressive cutoff of power use.

Aviation Data Management

The Flight Engineers and the automation systems both need to understand borders and constraints regarding any airspace with Loon-relevant attributes. This data is incorporated into the fleet management system and is viewable on the Smallworld UI. In many cases, this is informational, for instance, showing Area Control Centers (ACC or ARTCC) and their Flight Information Regions (FIR). In other cases, it's critical and needs to be monitored actively, such as no-fly zones. Where there are navigation implications, this same data also informs the navigation automation system to enable it to avoid no-fly areas, no-landing areas or other areas where Loon had particular navigation considerations.

Often, the restrictions or constraints around such airspaces are altitude (or Flight Level) specific so this is also indicated. Many of these areas are defined by ICAO, FAA or other local regulatory agencies and are relatively static, potentially lasting years, while others change status based on Loon discussions with local regulatory agencies. Still others were very temporary, for example, storm-avoidance no-fly zones that are automatically created by the weather hazards processing pipeline as storms form and are automatically removed as storms dissipate. For more information, see "Storm Avoidance" on page 290.

Aviation Notice to Airmen (NOTAMs) were also collected every 30 minutes and parsed to be displayed as polygons on the map. This helped Flight Engineers and the system to stay aware of changes in Airspace status, rocket launches and other dynamic airspace information.

Wind and Weather Data Ingestion, Management and Monitoring

Loon continually downloaded, ingested and utilized forecasts, live nowcasts and historical wind and weather data. This is described in "Wind and Weather Data" on page 243.

Fleet Navigation and Orchestration

Getting the individual balloons to steer to their targets and orchestrating the full fleet's motion was one of the most difficult but most enabling functions of the Loon software. This is described in "Navigating on the Wind" on page 255.

Network Management Systems Integration

Loon's service network was roughly divided into backhaul and access-layers. The backhaul systems included the ground stations, CommsNode and B2X systems on the vehicles, and was managed by the Minkowski TS-SDN software. The access-layer included the LTE systems on the vehicle and the Loon core network, both of which were handled by the Airstream and LTE management software.

These network management systems were integrated into the overall Fleet Management System, which, among other functions, provided a command and telemetry channel between those network management systems and their respective network elements.

Web UI (Smallworld)

The extensive Smallworld functionality allowed Flight Engineers and other Loon personnel to supervise the fleet and control the automation systems. Smallworld was also a critical tool for debugging and optimizing the many software, electronics and mechanical systems across Loon. Smallworld functionality is described throughout this chapter.

Additional Web UIs

It was important for Loon to provide visibility into its operations for external entities such as Air Traffic Control and this was typically provided through other web-based user interfaces, similar to a minimal Smallworld.

Integration / Interfacing with Simulation infrastructure

Loon used simulation of detailed system models and environments for several key objectives:

- Continued ensemble forecasting of the future positions of the active fleet– used to quantify likelihood of entering any special constraint areas in order to adjust steering automation to reduce likelihood, and to provide alerts to flight engineers and others.
- Perform detailed business analysis of potential opportunities, most importantly, understanding potential confidence and availability for a range of fleet sizes for different proposed service areas.
- Perform detailed systems engineering trade studies and optimization exercises to refine whole new vehicle candidates or proposed changes to specific subsystems.
- Train ML models to navigate balloons.

These required the simulation to consider the effects of all the navigation, health monitoring, storm and no-fly avoidance, and network orchestration and optimization software as well as the hardware and environmental sensitivities of the vehicle. The chosen way to accomplish this was to construct the simulation to make direct use of the majority of the production Fleet Management System software working with a simplified software model of the hardware and avionics. This ran against a modeled stratospheric environment, based on historical wind and weather data or forecast-derived ensembles of future wind and weather.

Wind and Weather Data

Loon systems could navigate balloons all over the world, reliably arriving at their destination, all by using the variation in winds found at different altitudes in the stratosphere.

It was a major challenge to design the algorithms and the software that could autonomously figure out how to use the winds, reliably and efficiently, to accomplish this navigation. This is discussed later in this section. It was also critical for the systems to have the best estimates of the winds along the routes and Loon's engineers had made significant innovations in this area as well.

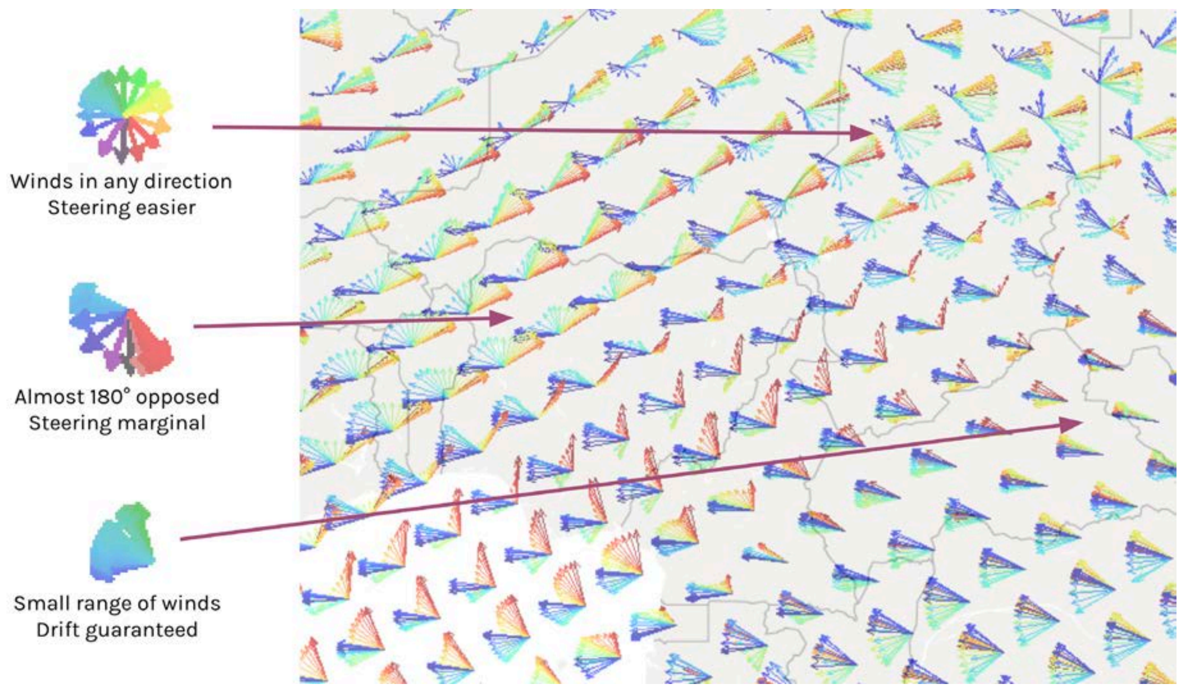


Figure 5-3 ***Wind cone illustration of the winds found across the altitude range available to Loon's vehicles.***

Data Sources

Loon used live and historical wind and weather data from several sources:

- ECMWF, the European Centre for Medium-Range Weather Forecasts,
- NOAA, National Oceanic and Atmospheric Administration
- NCAR, National Center for Atmospheric Research
- BCI, Basic Commerce and Industries, partner to NCAR to publish NCAR-originated data
- NASA, National Aeronautics and Space Administration
- Naval Research Laboratory
- Radiosonde and AMDAR (commercial aircraft) feeds

Most weather data sources are, as expected, focused mostly on ground-level and near ground level; few go into the stratosphere and fewer still cover the resolution in the stratosphere preferred by an operation such as Loon. Loon did use near-ground data for launch and landing operations and balloon-to-ground radio attenuation calculations but otherwise used high-altitude data, especially wind speed and direction, ambient temperature, upwelling infrared (coming from earth's surface), and weather hazard data (cloud-top height, and convective diagnosis).

Table 5-1 lists the real-time data sources Loon software used for navigating and managing the active fleet.

Table 5-1 Real-Time Data Sources for Navigation and Active Fleet Management

Source	*Use*	*Data Used*	*Resolution*	*Updates/ Range*
ECMWF HRES Forecast (Integrated Forecasting System)	Primary forecast	• Wind speed • Heading, • Temperature • IR • Lightning • Tropical cyclone info	0.5° x 3 hrs x ~370 Pa (6 hrs for days 6-10)	12 hrs/10 days
NOAA GFS (Global Forecast System)	Secondary forecast	• Speed • Heading • Temperature • Infrared, • Storm data	0.5° x 3 hrs x ~2500 Pa	6 hrs/16 days
NCAR / BCI (Global Weather Hazards project)	Nowcast of weather hazards	• Bad weather (CDO) • Cloud-Top Height (CTH)	0.06° resolution valid ± 3 hrs	15 mins/Now only
Loon's vehicle observations	Error correction to forecasts	• Wind speed • Heading	100 m x 50 Pa	~1 min

Table 5-2 lists the historical weather data Loon used for simulation and analysis.

Table 5-2 Historical Weather Data Used for Simulation and Analysis

Source	*Time range*	*Resolution*
ERA5 (ECMWF)	1979–2018	0.5° x 1 hr x ~370 Pa
ERA-Interim (ECMWF)	1980–2015	0.5° x 6 hrs x ~2500 Pa
GFS Analysis (NOAA)	2013-07-03 to present	0.5° x 3 hr x ~2500 Pa
And, stored historical versions of the above		
NCAR / BCI (CDO and CTH)	~ 3 yrs	0.06° x 15 mins
ECMWF Forecast	2016-07-08 to present	0.5° x 1 hrs x ~370 Pa
NOAA GFS	2016-07-08 to 2020	0.5° x 3 hrs x ~2500 Pa

Meteorological data was downloaded by an automated data pipeline in the source's standard format, such as grib, netcdf, grib2, or bufr. These were then converted over a number of steps into the final format required by Loon.. This data was served to other software clients with a common WindClient API, which used a merging of the ingested forecast data with live balloon observations, and, for some simulation uses, a constructed representative noise signal.

The wind data falls into several important categories:

- Directly-observed data: nearly all of the direct wind measurements that were relevant to Loon were from Loon's own balloons.
- Forecast data: best view of what will happen in the future
- Analysis or reanalysis data: best view of what did happen in the past
- Nowcast data: best view of what is happening right now

Keep in mind that there are very few direct measurements of stratospheric weather so the vast majority of the data we downloaded is the result of extensive processing of very sparse input data.

Wind Information (In)Accuracy

It was important for Loon to use the highest accuracy and highest resolution wind data available and so we analyzed all sources, comparing against its own observations of actual wind at altitude (ground truth). The error between the downloaded data (forecast or reanalysis) and the Loon-measured values are expressed in Error Vector Magnitude (EVM, also known as Vector Magnitude Error). This is the magnitude of the vector difference between the predicted wind vector (direction and speed) and the actual wind vector. The following charts compare the EVM for the ECMWF and GFS forecasts versus Loon's direct observations, first for the tropics and then for temperate regions.

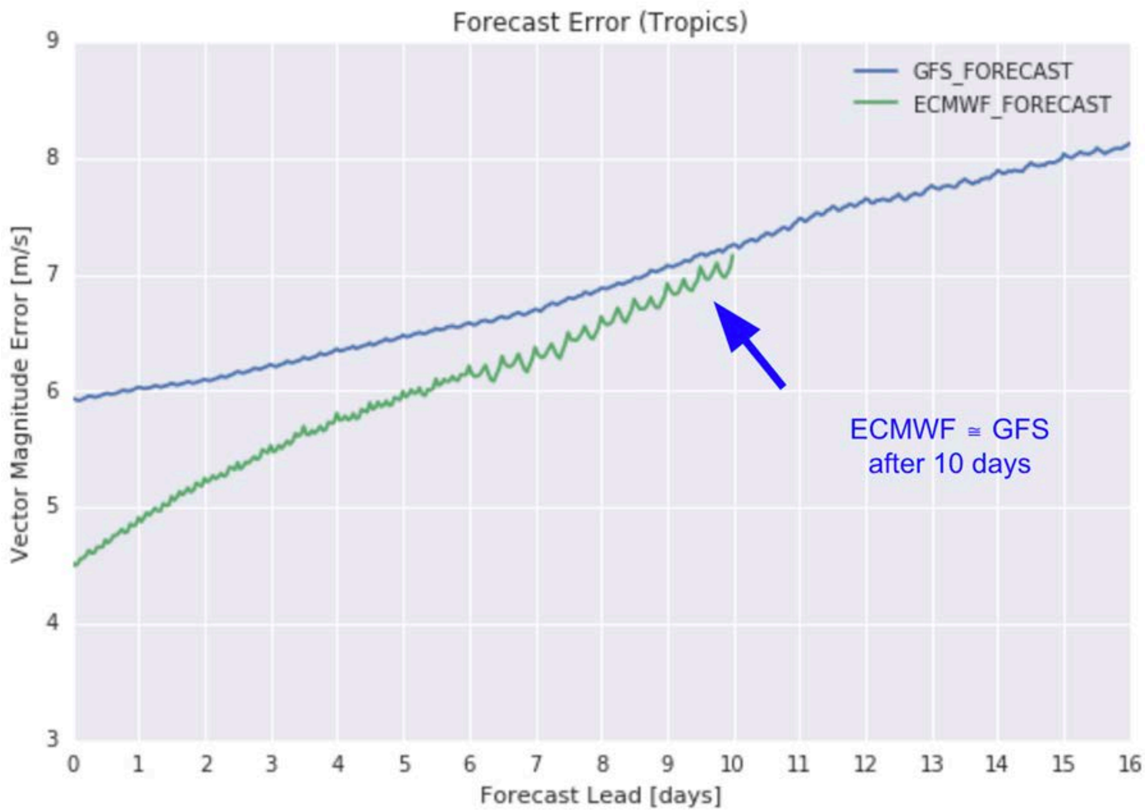


Figure 5-4 GFS and ECMWF wind vector magnitude error for the tropics (equator).

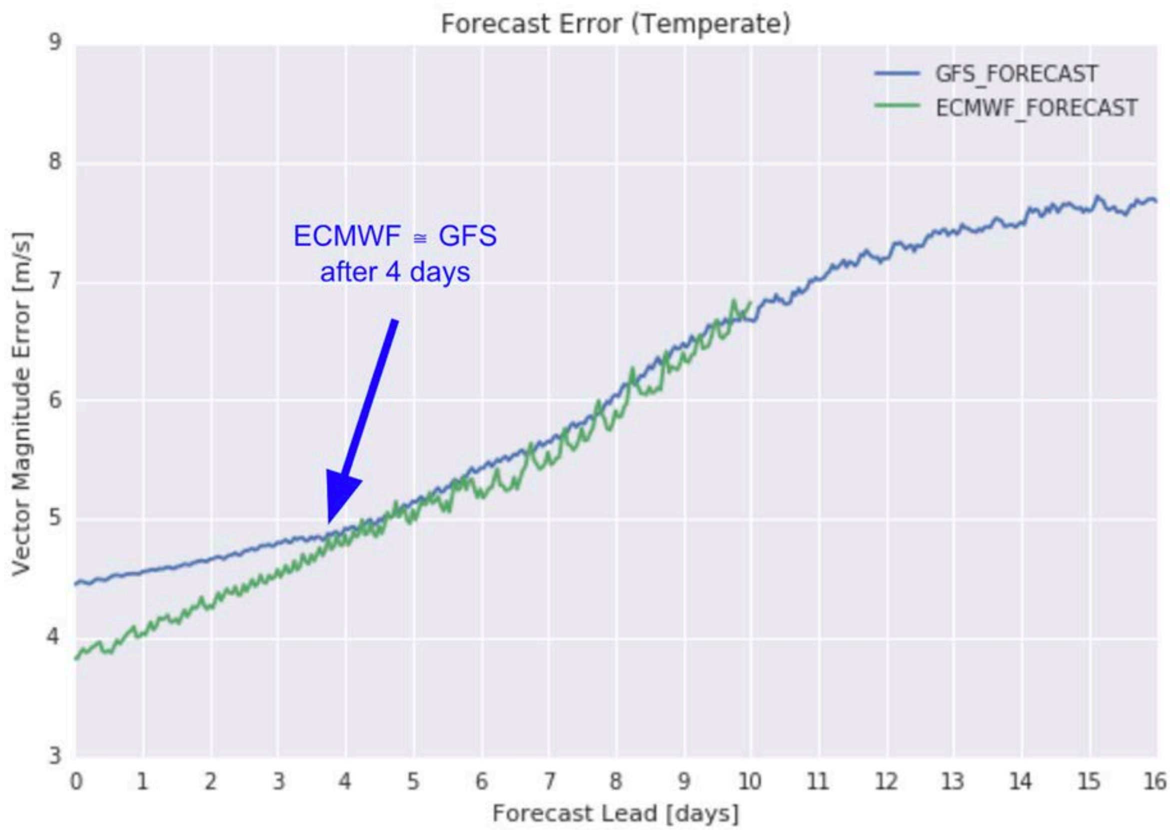


Figure 5-5 ***GFS and ECMWF wind vector magnitude error for temperate regions (far from equator).***

As seen in Figure 5-4 and Figure 5-5, the forecast error observed from Loon's primary data source (ECMWF) increases roughly linearly from around 4.5 m/s mean error for zero lead time (now) to 7 m/s for the 10-day forecast. The clear difference in forecast accuracy between the ECMWF and the GFS forecasts was one reason that Loon chose to move from GFS to ECMWF several years ago.

ECMWF's data was also available for many more altitude levels within Loon's operating range, as shown in Figure 5-6.

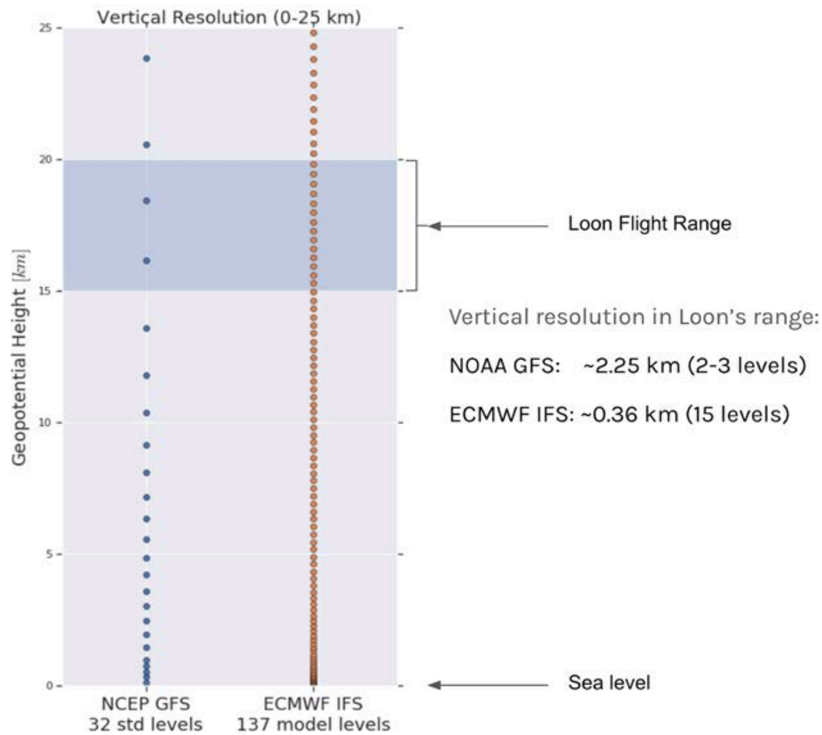


Figure 5-6 ***ECMWF data and NOAA GFS data in comparison.***
In addition to better accuracy, ECMWF data has many more altitude levels in Loon's altitude range (15 levels vs. 2-3 levels).

Incorporating Direct Observations–WindGP and WindMixer

For real-time operations, Loon created a piece of software infrastructure called Wind Mixer that stores recent observations from balloons and fuses them with wind forecasts to create a more accurate estimate of winds aloft. While many algorithms can be used to build this fusion, in production Loon settled on an overlapping lattice of gaussian processes (GP). Our specific implementation of each GP was dubbed a WindGP, which corresponded to the collection of kernels, hyperparameters, and representations of the data we found to maximize forecast skill.

At a broad level, the WindGP weighs balloon data more heavily closer to the sample location and time. The farther from that location, altitude and time, the weight of balloon data decreases and the weight of the prediction data increases. WindGP also calculates a confidence estimate, which also slides from high-confidence for recent direct observations, to low-confidence for locations far away from direct observations, either in space or time. This confidence value was used in the Steering Controller algorithms to drive their exploration of low-confidence altitudes, to look for potentially favorable winds. It was also used in the long-range navigation algorithms to value short-term forecasts over long-term forecasts.

An example of WindGP fusing measurements into the meteorological forecast field is shown in Figure 5-7.

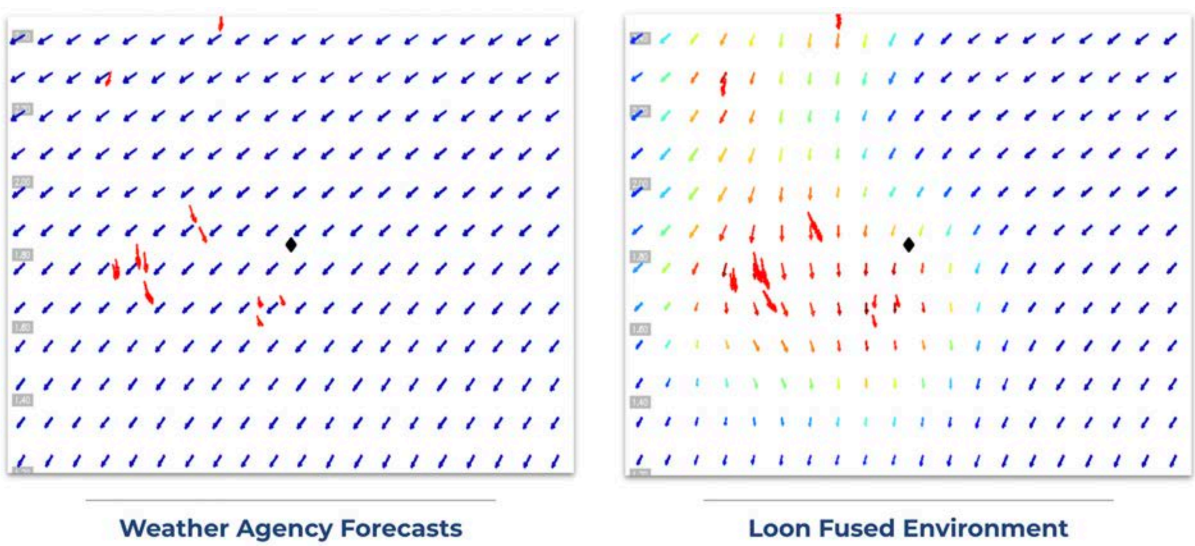


Figure 5-7 ***Wind data fused by weather agency forecast data and Loon's own measurements.*** Red arrows in the left diagram indicate Loon-measured wind vectors, with background purple arrows being forecast, and in the right diagram, the color indicates how confident the GP is in its combined forecast.

The resulting error correction provides a short-term improvement of over 75% as shown in Figure 5-8.

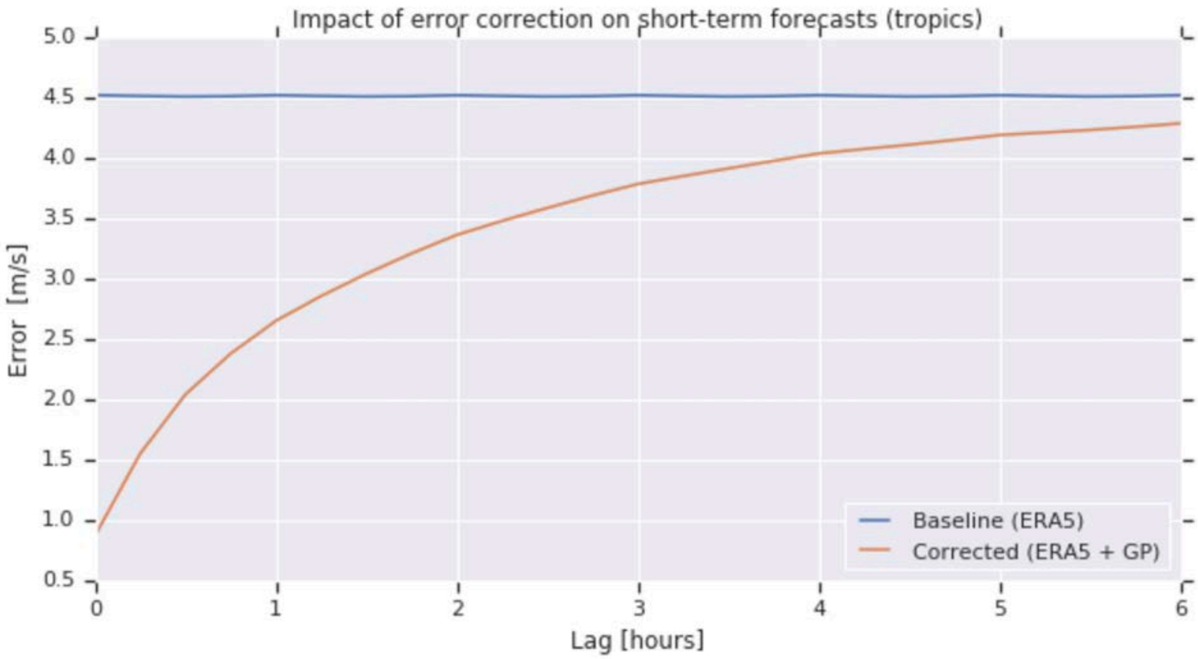


Figure 5-8 ***Utilizing WindGP reduces the error in the immediate forecast by over 75%.***

Wind Modeling and Wind Noise

In order to simulate fleet behavior, the simulated environment needs to present both forecasts and actual winds to the steering and navigation software in the model. If the same wind data was presented for both, then the steering quality would improve dramatically because that would magically make all of the forecasts 100% accurate.

Loon's direct measurements of the winds are very useful for presenting more accurate wind information to the flights in that area at that time, but this data is not available for simulating over past historical periods, or current periods but in areas where the balloons have made no recent measurements. For simulation over past periods, Loon uses historical reanalysis data from the ECMWF ERA5 data set. This is an estimate of historical weather and is generally close to the actual weather that happened at that time and place because it is based on actual historical observations. However, it still can differ from the actual weather, especially in the stratosphere, because those historical observations are from sensors on the ground, relatively sparse weather balloon measurements, and commercial aircraft, which are very sparse at high altitudes. Because of this scarcity, the reanalysis data for Loon's flight levels does not consistently reflect the real structure or values of the winds.

In addition to the errors, forecasts (estimates of the future) and reanalysis (estimates of the past) do not capture the high frequency effects that happen in the real world. For example, the estimate may show, for a particular altitude layer, winds that continuously increase in speed and slightly change direction as you go across the country of Kenya but the actual winds would vary much more than that, increasing rapidly, then decreasing, holding steady for a while, before increasing again. Likewise, the directions would often vary more than the estimates. Essentially, both forecasts and reanalysis data do not have enough input data to create these dense variations and so they present a highly filtered, smoothed view.

To improve the realism of the wind estimates presented to simulation, wind noise is added to create some of these real-world effects; to un-smooth the data. This augmented wind data won't exactly match the actual winds but will be more realistic in terms of overall variability, structure and error.

Historical Simulations

For simulations across historical periods, the simulation infrastructure uses ERA5 historical estimates as the forecast data, because it is very filtered and smooth and so resembles (in structure and variability) forecast data. The system then adds wind noise onto that forecast data to create the "actual" winds. As described, this creates the higher-frequency (less smooth) variations in wind speed and direction that are seen in the actual world, and, as importantly, have a large impact on Loon's steering algorithm performance. Since the goal of simulation is to see how well those algorithms work, this was an important factor.

Oracle Simulations of Active Fleet

In addition to using wind noise to create more realistic winds for simulations over historical days, wind noise is also used to generate a more robust navigation plan and estimate the likelihood of a plan succeeding. It does this by adding wind noise to the forecast data and doing ensemble groups of simulations using these noisy forecasts, where each simulation in the ensemble uses a different randomization seed for the random wind noise. This system was called Oracle.

This creates a variety of projected flight paths for each vehicle, then, for presentation, a polygon is wrapped around those paths to give the flight engineers, and the navigation systems, a sense of where the balloons might end up in a day, or two, or a week.

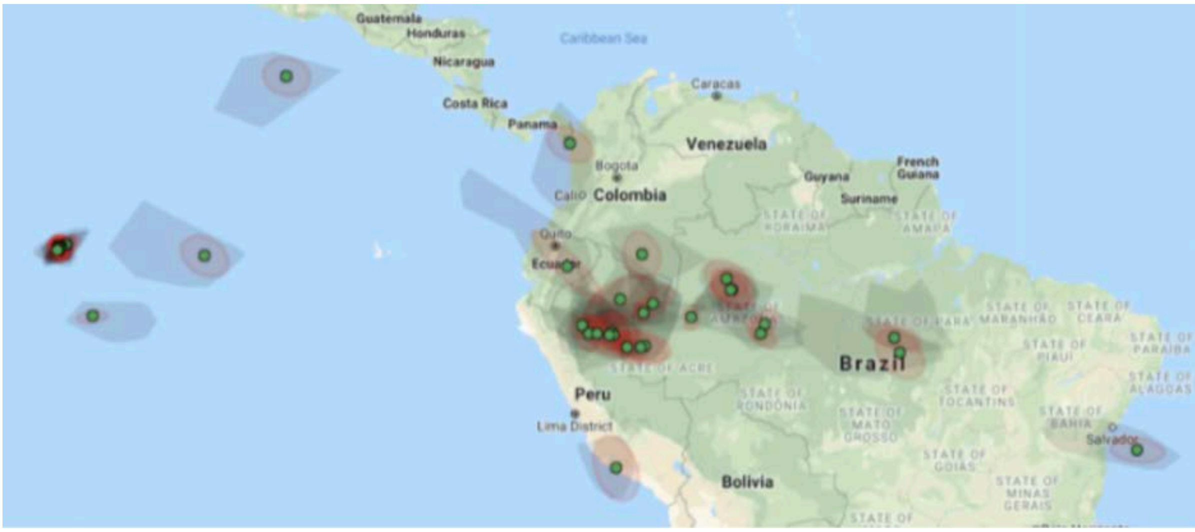


Figure 5-9 ***Smallworld's Oracle forecast.***
For each balloon in the fleet, it shows a range of potential balloon locations over a future time period. Oracle performs an ensemble of simulations using a variety of wind noise seeds to generate this volume of potential paths, rather than a single projected path.

Oracle was used by Oregon Dispatcher to evaluate the chances of success for a particular task being considered, or the chances that a balloon might enter a no-fly zone when performing that task.

Improving Long-Term Wind Forecast Accuracy using Analog Ensemble

The analog ensemble (AnEn) technique described by Loon[1] is a statistical post-processing method to improve numerical weather forecasts, based on historical predictions made by the same forecasting system. Applying the AnEn technique to the ECMWF forecast data reduces errors up to 12% for wind speed and 20% for wind direction, with larger improvements coming at longer lead times. One of the challenges of applying this technique was being able to perform the procedure at the operational speed required.

This challenge was initially solved by utilizing a distributed computing implementation on the Google cloud platform then later by utilizing a deep neural network to perform the post-processing.[2]

1 https://medium.com/@salcandido/improving-wind-forecasts-in-the-lower-stratosphere-87a8536cfb52

2 https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL089098

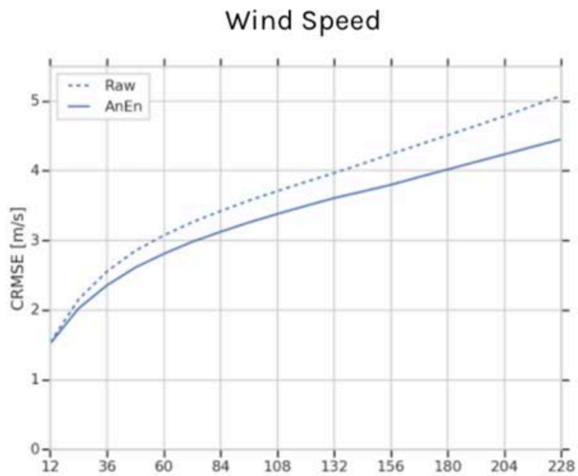


Figure 5-10 ***Comparison of raw versus post-processed ECMWF data error in wind speed and direction.***

Weather Hazards Data

Loon balloons operate in the stratosphere above 50,000 ft but are still susceptible to turbulence and electrical activity from storms well below the balloon, which can cause mechanical and electronics hardware failures. Tall clouds can also cool the lift-gas, causing the balloon to lose buoyancy.

To combat this, Loon's navigation algorithms incorporated two nowcast data streams generated by NCAR (National Center for Atmospheric Research): CDO (convective diagnostic oceanic) and CTH (cloud top height). CDO defines a region of convective (that is, storm) behavior along with a likelihood of convective hazards. CTH on the other hand provides the height of the top of the clouds. CDO and CTH are provided by BCI Systems Software Engineering and are updated every 15 minutes. See "Storm Avoidance" on page 290 for more details on how Loon used this data in our production systems.

Navigating on the Wind

History

It was initially proposed that the balloons would drift, circumnavigating the globe and small amounts of propulsion and altitude control would be used to space them uniformly. Loon would be required to deploy and continue to supply a global constellation of tens of thousands of balloons to provide service to any specific region. By 2015, significant progress had been made on understanding and modeling the wind patterns and Loon determined that the winds in the tropics were sufficiently diverse to keep the balloon in a specific latitude band. By 2017, Loon's algorithms had advanced enough to automatically navigate balloons to desired regions, stay within those regions for extended periods of time and, if blown away, navigate back to those regions, even if it means looping around an entire continent or all the way around the world.

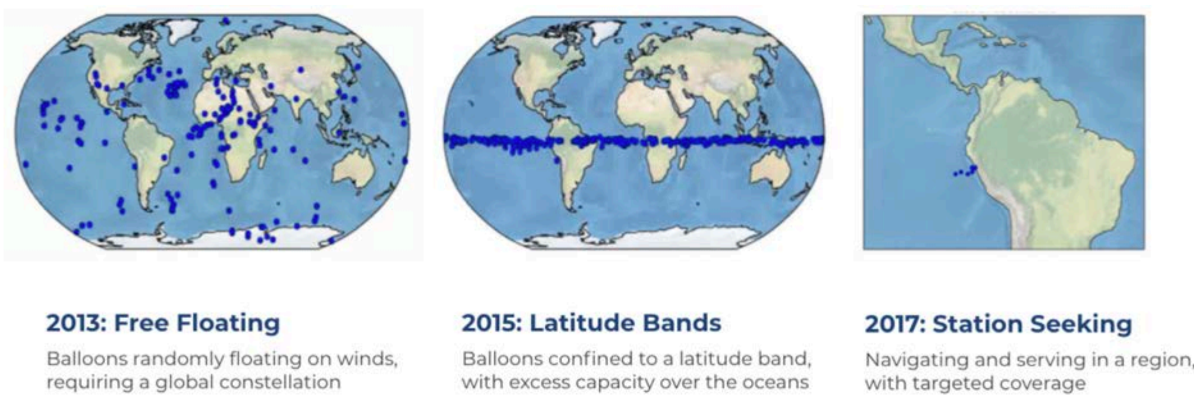


Figure 5-11 ***Loon Fleet Balloon Placement Improved Over Time Using Modeling and Targeted Coverage.***

Since that time, continued advancements have been made, both to the ability for a single balloon to reach and stay near a target but also in orchestration of the entire fleet of vehicles to optimize their arrival and departure times over a service region, allowing them to maintain continuous coverage.

Loon's use of different winds at different altitudes to steer balloons makes the flight paths highly dependent on the actual winds but, because the steering controllers use forecasts to make the steering plan, the success of that plan depends on the accuracy of the estimate of the winds.

Loon made innovative advances in improving the accuracy of wind forecasts, such as the Analog Ensemble approach, but the majority of the learnings and innovation that enabled Loon to navigate the balloons so successfully was in the algorithms invented and the software implementation of those algorithms. These systems were designed to live completely within this uncontrollable and unpredictable reality but to make the best use of the few forecast signals that were available, even if the confidence in those forecasts was low.

Navigation Overview

Loon's expectation of having thousands of balloons flying in the stratosphere made it important for the balloons to be navigated to specific targets automatically, without manual human intervention. Loon did employ a small team of Flight Engineers dedicated to supervising the fleet and fleet management systems (FMS), but this team focused on exception handling and improving the systems, rather than active navigation. For the most part, the fleet navigation systems flew the fleet.

Loon's fleet navigation and orchestration software has three major layers, starting at the bottom:

- Locally-acting Steering Controllers, which are local planners that use nearby wind information or heuristics to control a single balloon trying to meet a specific (local) objective. For instance, the Steering Controller for a balloon may know the target is 100 km due east, and it finds and uses winds that are closest to that direction to move closer to that target. It's oblivious to the winds beyond its very local view and ignorant of any higher-level objective the system may have for that vehicle. The Steering Controller is the software that decides what altitude to move to, or what direction and speed the propeller (if present) should use. Because the Steering Controllers are only accessing local winds, they are only useful for very local objectives.
- A long-range navigation system, named Cartographer, which uses a global view of the wind environment and looks weeks into the future to generate global Cartographer Maps for a specific target. These indicate, for every spot on the map, the best direction a balloon at that spot should travel to meet that map's objective (e.g., southern Kenya). In many cases, the direction indicated

will be in a very different direction than the target itself: though Kenya may be 1000 km directly east, the map may point southwest in order to catch a much faster or more reliable wind to that area. The Steering Controller for a balloon doesn't need to look at winds along the route and doesn't even need to know its ultimate target, it just needs to understand that the map says, "go southwest" and try to find local winds heading southwest.

- Fleet-level orchestration algorithms, called Dispatchers, which determine which target each balloon should be trying to reach and at what time, in order to optimally use the balloons available to provide near continuous coverage over the targeted service areas. For instance, when winds are all aligned at the service area and balloons will be blown right through, the Dispatcher will assign each balloon to arrive at the service area at a different time so that the balloons file past, one after the other, keeping the area covered.

To get a better understanding of how this all works in practice, imagine a fleet of 20 balloons allocated to serve southern Kenya. As a new balloon is launched from Puerto Rico:

- The Dispatcher sees that the new balloon is capable of serving Kenya and assigns that balloon to a Cartographer-generated southern Kenya ASAP map. That map is generated from forecasts covering the next few weeks, and the map shows, for every point on the globe, which direction the balloon should fly in order to get to southern Kenya as soon as possible.
- Periodically, the Steering Controller looks at the assigned map and decides whether to ascend or descend in order to catch winds that take it in the indicated direction. Often, the Steering Controller has little information about winds above or below the balloon's current altitude so it will occasionally explore other layers to determine if any are better than the layers it already knows.
- As the Steering Controller explores new layers, those new wind observations are taken as the best prediction for that location, altitude and time (now) but, using WindGP, that wind measurement is also mixed into the predicted wind vector for layers just above and below the current altitude and across nearby locations as well as the near future for these nearby points.
- Only rarely would the Steering Controller find a wind direction that is very close to the desired direction and, given the lack of confidence about the future of winds, there will always be uncertainty about which layer is the best to choose. Different Steering Controller algorithms would choose layers differently.

- As the balloon gets closer to Kenya, the Dispatcher is monitoring the existing fleet and the forecasts and weighs the probabilities of each balloon hitting their targets during particular time slots and from this, decides to assign the new balloon to a time slot in 2 days, by tasking that balloon's Steering Controller to follow a Cartographer-generated Arrival Time Map (vs. ASAP), built with that specific time slot and Kenya as the objective.
- The Arrival Time Map indicates to the Steering Controller which headings are most likely to have the balloon arrive at the destination during the assigned time slot. In many cases, the preferred direction will take the balloon on a route that takes longer to get to the target than an ASAP route. For instance, if the balloon is only a half-day away from the targeted area when the Dispatcher assigns it to an Arrival Time Map for 2 days out, that map at the current balloon position, will take the balloon towards a route that will take 2 days to get to the targeted area.
- As the balloon's assigned time slot nears, the balloon itself should be nearing the service area. At some threshold, the Dispatcher then assigns the balloon's Steering Controller to seek toward and maintain position around a specific target within the service area. This Steering Controller is now operating under a location, not a direction objective. As it gets closer or passes by the target, the direction between the balloon and the target will change, causing the Steering Controller to move to different wind layers.

Then it gets complicated. These basic mechanisms have significant complexity built in to handle the many problems that can be caused by highly random winds that are difficult to forecast with decent accuracy.

Steering Controllers–Local Planners for steering

The goal of the Steering Controller design is to efficiently steer the balloon toward its objective, in some cases that objective is to maximize proximity (closeness) to a target location (location objective) and in other cases, it is to move in a specific direction that's determined from a Cartographer ASAP or Arrival Time map (direction objective). Loon developed and retired a number of Steering Controllers through its history, learning and improving along the way. Not all of those Steering Controllers could operate in both objective modes: direction- and location-seeking.

At the highest level, Steering Controllers take as input:

- Balloon telemetry
- Local wind field data, incorporating recent direct measurements of winds by this and other (nearby) balloons through the WindGP + WindMixer process
- An objective (as above)

The output is a series of commands to move the balloon to different altitudes.

There are a number of problems and constraints that Steering Controllers need to contend with:

- Even for the best weather services, the forecast error (for stratospheric winds) is too high to be able to plan deterministically, and this impacts all aspects of the Steering Controller behavior. To address this, it was critical for the balloons to occasionally explore the winds throughout the accessible altitudes rather than trust the forecast and then to exploit that updated wind knowledge. This *explore and exploit* functionality was required in all Loon Steering Controllers and their different algorithms approached this problem differently.
- Balloons cannot teleport instantly to different altitudes and instead take about 2 to 4 hours to traverse their full range. Loon referred to this vertical speed as steering agility. Better steering agility can be attained with more powerful ACS pumps, but gets worse for larger balloon designs, if the ACS capability remains unchanged.
- To descend, the balloon required not only time but also significant power for the ACS to pump sufficient air into the balloon and again, larger balloons required more energy to descend the same distance. Ascending takes no energy but does still take time. The different Steering Controller designs took the time and power dimensions into account in different ways.
- There are changing constraints that affect which maneuvers are possible or risk free, due to the changing temperature of the balloon gasses and ambient and, potentially, due to degradation in the envelope or ACS.
- The optimal behavior of the Steering Controller may change depending on the distance to the target (if it's location-seeking). Far away, it may make sense for the Steering Controller to find the winds with a direction that is closest to the target direction but may also be favoring fast winds. As the balloon nears the target, slower wind speeds may be optimal. The Steering Controller must also smoothly interpolate between behaviors in such cases.

- As the balloon passes by the target, the desired direction will change rapidly, potentially encouraging a Steering Controller to change altitudes rapidly trying to exploit different winds. This may not be possible, due to the finite vertical agility and limited energy available.
- There are storms below Loon's operating altitude that, through turbulence and electromagnetic activity, can potentially damage the balloon, sometimes ending the balloon's useful service life. In addition, there are no-fly zones of various types and restrictions but in all cases, the navigation and steering systems avoid entering these areas if possible.

Loon's Steering Controller designs fall into three broad buckets:

- Tree Search, or optimistic deterministic planners
- Seeker-based controllers
- Controllers trained using Deep Reinforcement Learning (RL)

Tree Search Algorithm

Tree Search planning explores the universe of possible move sequences that the balloon could make, simulating each possible sequence to score how well that sequence met the Steering Controller's objective. There are several disadvantages to using a tree search algorithms in Loon's situation:

- The underlying random behavior of the winds themselves, the limited visibility into the future of this random behavior and inability to determine the actual uncertainty, meant the local planner often took immediate action based on information that would almost certainly change. This could often be mitigated but doing so required a significant computational premium above simpler approaches that created similar solutions.
- The optimistic, deterministic nature of the tree search approximation caused the steering controller to underestimate the uncertainty of the world and choose paths and solutions that were not robust to small perturbations in the wind field. This would often cause problems in locations where navigation needed to be precise.
- For computational reasons the search was typically conducted using a simplified model of the balloon. This could be unwieldy in production as yet another thing to manually tune as new aircraft became available, and different models needed to be carefully managed for everything currently in flight and in simulation. In contrast, our more modern approaches of control (like deep RL) were designed to avoid engineering time to manage new systems.

- A well known and well studied concern about tree search is about the exponentially expanding search space. This happens as the planner looks farther into the future or broader into different possible (uncertain) outcomes. This is not unique to Loon.

These problems constrained the useful size of the planning window and the navigation objectives that Loon used it for. Nonetheless, tree search was a tool we used since the early days of Loon and continued to leave running in production as we shifted development focus onto Sleepwalk, the RL-based solution.

Very Simple Planner (VSP)

VSP was one of Loon's tree-search-based Steering Controllers. VSP tries to find a control plan over the next eight to ten hours that maximizes progress toward its location or direction objective while considering relevant balloon state and capabilities, such as the amount of power available and the ascent and descent rate of the vehicle. Because it was a tree search algorithm (see issues above), it constrained its planning window to this short period.

Despite the disadvantages of tree-search for Loon, VSP's ability to consider the vertical speed of the vehicle and the available range of descent, due to constrained energy, did provide advantages over earlier Steering Controllers that ignored these factors.

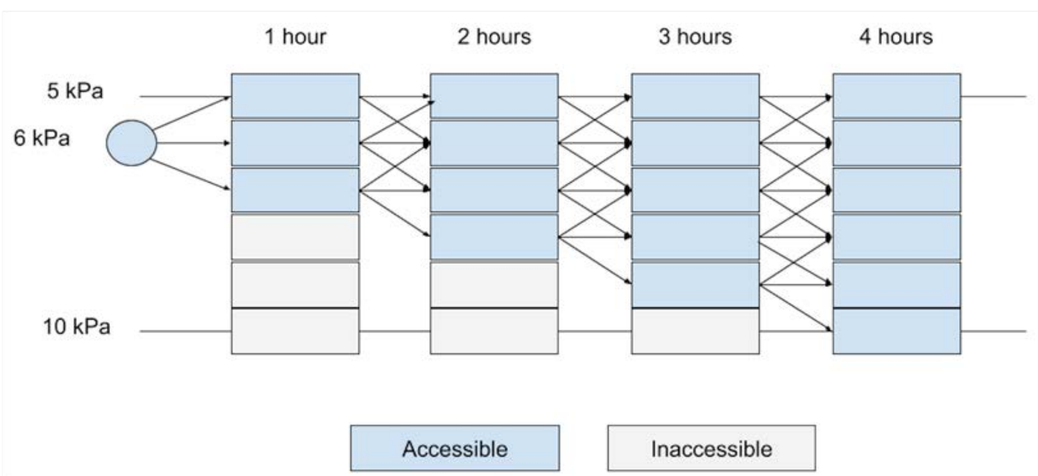


Figure 5-12 ***For a balloon with vertical agility of 1 kPa/hr, most pressures are inaccessible within the first time stride.***
The full range only becomes available after 4 strides. VSP's ability to consider this constraint was an advantage in some circumstances.

Seeker Algorithm

An alternative to a brute-force tree search approach is to use a more hand-tailored, engineered approach, such as Loon's Seeker algorithm, which significantly advanced Loon's steering capabilities and was the basis for several important Steering Controllers over the years.

Seeker's objective is to approach and stay near a target. Seeker looks at the altitude sounding (winds available at reachable altitudes) and evaluates two factors:

(1) *what is the expected heading error at a given altitude* and

(2) *how confident are we in that heading estimate.*

The heading error (see Figure 5-13) is the angle between the direction the balloon is expected to go if it continues in a straight line and the direction from the balloon to the target position. The confidence is based on how recently we've taken a measurement that is "close" to that point in latitude-longitude-altitude-time. One might imagine the balloon's current heading is known with high confidence, but the expected heading a few thousand pascals away is not as certain.

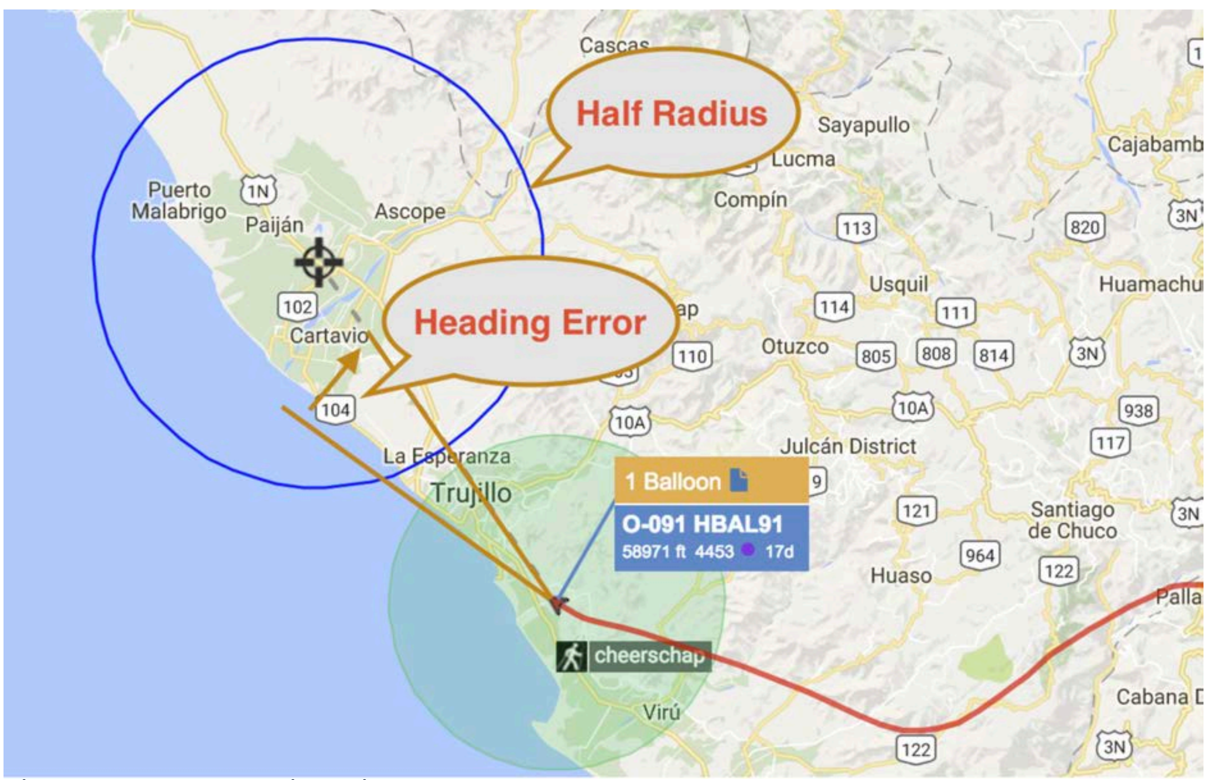


Figure 5-13 Seeker in action.

Seeker uses a Seeker promise function to rank the desirability of altitudes the balloon can reach. This includes the possibility of staying at the current altitude.

The best promise is found at altitudes having a small heading error with high confidence. (Given equal confidence, promise increases monotonically with decreasing heading error.) At some point the heading error is large enough that points with low confidence, regardless of expected heading, are considered more promising. This basically creates an incentive for the balloon to explore areas of the altitude range where something good might happen when we don't have a sufficiently good high certainty option.

There are more factors involved in the promise function, but when the balloon is sufficiently far from the target these are the main terms. The algorithm chooses the altitude with the maximum promise at every point in time. The result is an emergent behavior which causes the balloon to constantly "seek" towards the target. (See Figure 5-14).

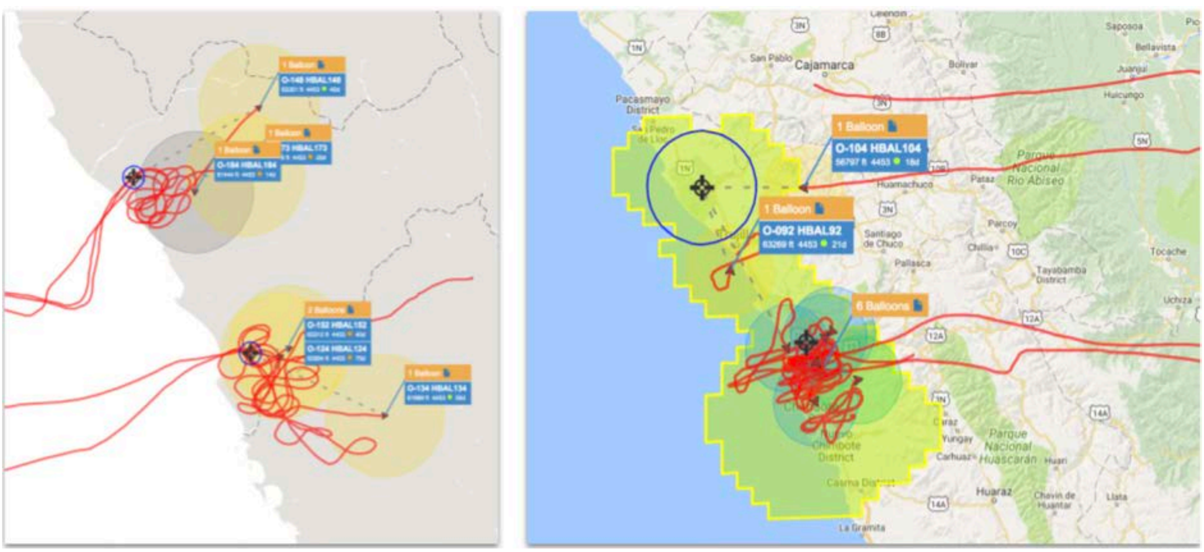


Figure 5-14 Station Seeker emergent behavior.

The above behavior makes sense to drive the balloon to the target when it is far away, but one could imagine this is not the optimal thing to do when it is very close to the target. The heading error could get quite large even though the balloon will only miss the target by a few kilometers (which would still be a big win with a 30 km LTE radius).

A good way to handle this issue is blending the promise from a heading-based objective function to a speed-based function. When the vehicle is very close to the target we say an altitude level has high promise if the balloon would be moving very slowly regardless of direction. Close is defined as the “half-radius” (see Figure 5-13), the circle where we value speed and direction equally.

The result is an emergent behavior that shows features of exploration (try to get more information about the winds and navigation options) and exploitation (fly at the altitude that is the best known alternative to achieve the goal). This mix was effective for balloon steering and so for years, almost all automated flight occurred under the control of a Seeker algorithm.

Seeker Promise Function

The Seeker decision rule chooses the action that maximizes the reward right now. However, the reward heuristic (or function) is engineered to induce some non-greedy behaviors (like exploration) as system behaviors.

The algorithm begins by determining the altitude range the balloon can utilize at the current time. It then breaks that altitude range into chunks (50 Pa works well) so that it can compute its optimization function on a discrete number of points rather than creating a continuous curve. This approximates a continuous function across the altitude range, but because much of the computation requires discrete steps, e.g., remote procedure calls to lookup data, discretization makes sense.

The Seeker algorithm runs each time the system has new wind data from an updated WindGP, based on this balloon’s and nearby balloons’ movements. Each time the algorithm runs, it chooses the discretized point in the altitude range that seems the most promising and instructs the balloon to move to that altitude. The information it receives from the movements changes its inference of winds, and by proxy, changes its guess which altitudes are good or bad.

A *promise* (*V*, for value) is computed for each altitude (*x*) according to the following equation:

$$V(x) = k[1 - C(x)] + [C(x) + \epsilon]G(x)$$

Where,

C is confidence, between 0 and 1

G is goodness, roughly between 0 and 1

x is the altitude

k and epsilon (ϵ) are constants

At each altitude (x) there are two important values: confidence (C) and goodness (G). The confidence ranges from zero for no-confidence, to one, if WindGP is fairly certain about the winds at that altitude. Goodness is also roughly between zero and one (but not in all cases) and represents how useful the winds are towards our goal. Specifically, it is an encoding of the different objectives that Seeker is using at that moment. Note that promise (V) is computed based on an estimate of goodness and not goodness itself. Actual (ground truth) goodness is unknown unless we can observe the entire wind field.

The first term of the equation incentivizes exploration. The lower the confidence, the higher the incentive. If confidence is low (zero) at some altitude this term encourages the decision rule to reward a visit to that altitude with a promise of k. If confidence is high (one) then the system has recently observed this altitude and gives no additional incentive to visit (beyond the following terms).

The second term of the equation incentivizes exploitation. If confidence is low it pays very little attention to an estimate of a high goodness value. However, if confidence is high it assigns the full goodness to the promise function. If this goodness value is sufficiently high the decision rule will chase this altitude and continue to exploit the known good winds.

The parameters k and epsilon govern the balance between explore and exploit. The constant k gives the threshold where something can be considered sufficiently good to settle, i.e., G(x) > k (for epsilon small) implies G and not C will dominate V at x. If goodness is well-known to be better than k at a point, Seeker considers that point more promising than pure exploration.

Epsilon helps guide the direction of exploration. It is denoted as epsilon because it tends to be a tiny number. Note when confidence is small then V(x) ~ k + epsilon G(x). This basically tells Seeker that everything is equal but Seeker should slightly prefer to first explore the regions where the unsubstantiated guess of goodness is highest.

Modifications

Since the initial version, many small tweaks were made to the Seekers. There are several worth covering.

The first is the altitude control change penalty. With this change, the promise function looks like the following:

$$V(x) = k[1 - C(x)] + [C(x) + \epsilon]G(x) + a_1 \exp\left(-a_2|x - x_0|\right)$$

This is identical to the equation above with the addition of a third term. The third term contains two constants a_1 and a_2 and is based on the distance to the altitude being considered (x), from the current altitude x_0. This term acts as a deterrent to making big moves through the altitude range.

The second modification was flattened confidence, which was introduced in response to the balloon often undertaking long exploration to learn about some very small portion of altitude range. This can best be described as C(x) no longer being the confidence exported by the WindGP (proportional to variance) at the altitude x, but instead being a function of all confidences above and below the balloon.

Specifically, flattened confidence is the average confidence between x (altitude of interest) and the balloon's current altitude (x_0):

$$FC(x) = \frac{1}{|x - x_0|} \int_{x_0}^{x} C(s)ds$$

FC(x) substitutes in the promise equation for C(x) in all terms.

The intuition is that we'd like to penalize moving through many contiguous points that have already been explored to reduce a small amount of uncertainty.

Further enhancements to Seeker were to maximize the time spent within or close to a target region, rather than to a target point. This better aligned with the actual objective of spending as much time as possible within a certain distance of a target.

Consider Figure 5-15. A “close to target” boundary defines the region we would like to keep the balloon within. The distance to the region within this boundary is considered zero, and outside it is the distance to the closest point on the target boundary.

For each altitude, the expected heading and speed are projected forward for a specific number of hours (a few). Figure 5-15 shows an example where we consider four discrete altitudes, but in practice many more are considered. Note that each line extends at a different angle (corresponding to heading) and is a different length (corresponding to speed). We consider the Goodness of a particular altitude to be the average distance to the region along this line, of course adding in the factor that also responds to confidence and causes the balloon to explore.

Figure 5-15 ***Seeker key elements.***
(Close-to-)Target region and boundary indicates an area rather than a point as objective. Different altitudes are valued by the average distance to that target region along an N hour projected path for the winds at that altitude.

Sleepwalk–RL-Trained Steering Controllers

Over the past few years, Loon has focused on developing a series of Steering Controllers trained using Deep Reinforcement Learning. The bulk of the work was not in writing the Steering Controller software itself, but in creating the software infrastructure that was used to train the Steering Controllers. This training was run over millions of simulated flight hours over a wide variety of conditions. Over weeks of training, these Steering Controllers learned what policy was best able to keep the balloons within a given radius of their target for as long as possible.

All future Steering Controllers were expected to be RL-based, using the Sleepwalk framework or similar.

Using RL essentially replaces the engineered decision rules used in prior Steering Controllers, with a system which learns through many iterations of a training process. While the referenced Nature paper only dealt with simple station keeping for academic purposes, our production controllers performed important additional functions, such as avoiding weather and restricted flight areas. Prototype controllers existed and were tested in flight for Cartographer map following, control of lateral propulsion systems, tighter position targeting and periodically visiting a target, and multi-balloon control.

The development of this approach stemmed from the forecasted increased complexity of Loon's navigation system. Additionally, with Loon increasing the number of vehicle types, deployment locations, and potentially expanding into multiple types of service, the required number of optimized controllers for each scenario would grow dramatically. Reinforcement learning controllers would manage higher levels of complexity across this wide increase of different use cases. It would also replace heavy modeling assumptions in the traditional Steering Controllers with data. This would pave the way to create better navigation strategies across the board for all Loon's systems and objectives.

The results were promising, as seen in Figure 5-16, comparing one of the first Sleepwalk controllers (cannelloni7), to StationSeeker. The vertical axis is the distance to target, with a thick dashed green line at 50 km.

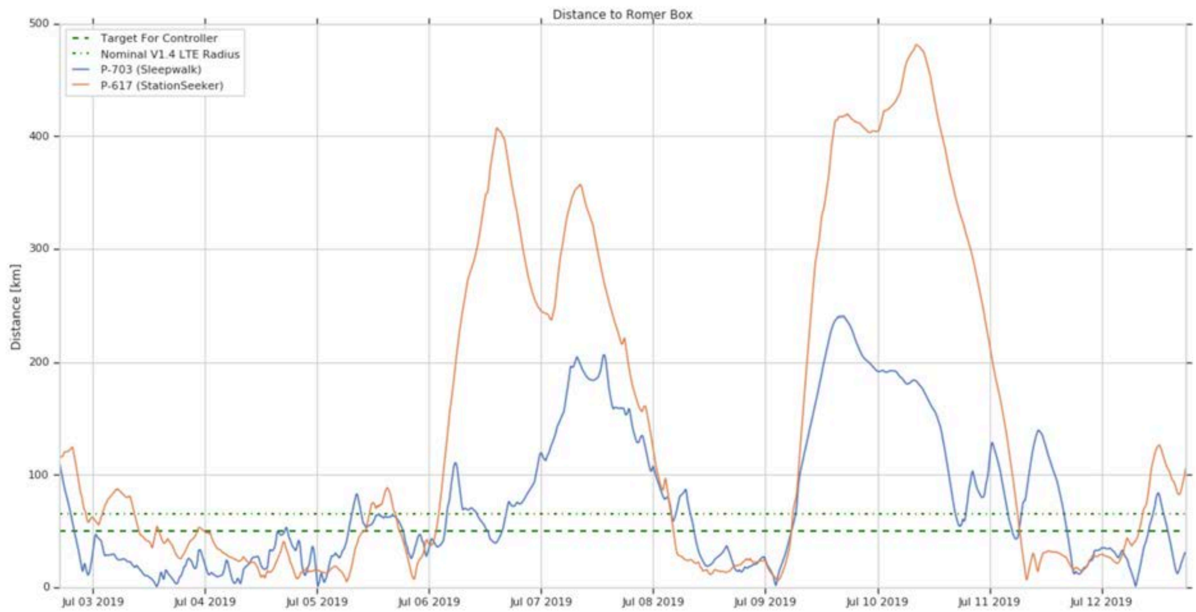


Figure 5-16 ***Early results of cannelloni7 vs. StationSeeker shows the pasta has better TWR50 results while avoiding the far drifts of StationSeeker.***

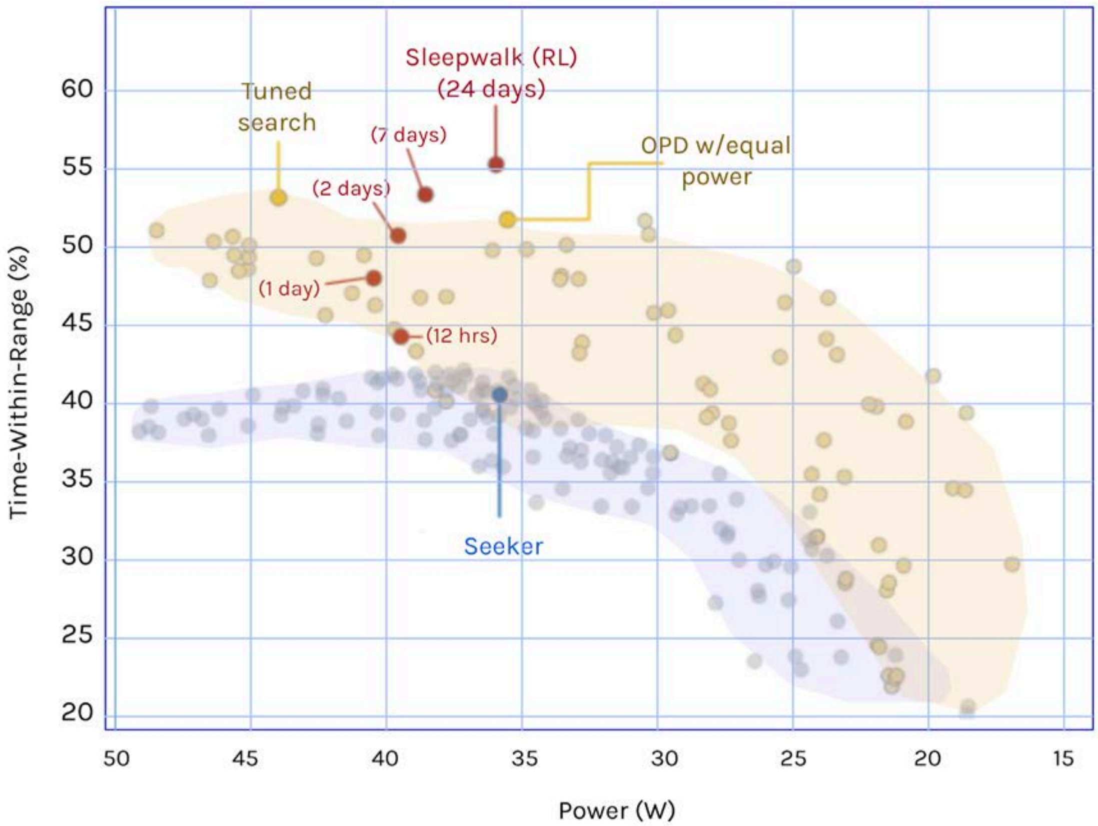


Figure 5-17 ***Published results comparing RL-trained against several other optimized steering control algorithms showing improved time within range (TWR) performance for RL-trained controller.***
Also note the increased performance over the course of training.

Some of the more subjective things were also interesting. On the left is the flight path of a balloon on the conventional StationSeeker while the right is the trajectory on Sleepwalk. The Sleepwalk flight path is much smoother as shown in Figure 5-18.

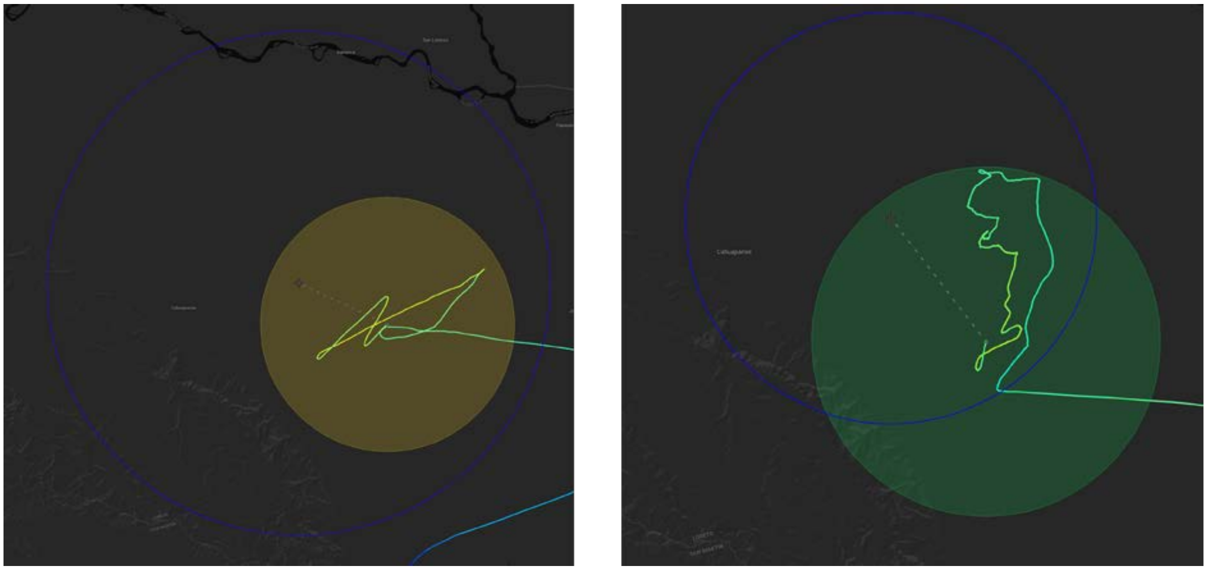


Figure 5-18 Balloon Flight Path Comparison: Conventional StationSeeker and Sleepwalk.

Cartographer–Long-Range Navigation

As described in the Navigation Overview section above, Cartographer is the system that generates the long-range navigation Cartographer maps, which are used by Steering Controllers as guiding maps to follow the long, circuitous paths needed to make their way to distant targets.

There are two flavors of Cartographer maps: ASAP and Arrival Time (note that Arrival Time maps were generally referred to as Cartographer Arrival Time, or CAT maps). Both of these are built for a particular destination region but the Arrival Time maps are also tied to a specific arrival time window. Both cover the entire globe and use S2 cells to identify the coordinates of the grid of points on the map. S2 cells are small areas, not points, so, in the following, the use of the term cell is typically referring to the location at the center of the S2 cell but may also represent the full S2 cell area around that point.

Note: S2 Cells are earth coordinates from Google's open-sourced S2 Geometry library, which Loon used to perform many spherical geometry calculations and manipulations. If you're not familiar with the S2 library, it's a powerful set of utilities for similar problems and an interesting read: https://s2geometry.io/devguide/s2cell_hierarchy.html.

On an ASAP map, each cell on the map conveys the number of days a balloon at that location will (most likely) take to arrive at that map's target, that is, the time-to-target. A Steering Controller assigned to such a map can determine the preferred direction (to get to the target ASAP), by looking at the time-to-target values for the cells around its current location and figuring out which available wind direction would lead towards the cell with the lowest time-to-target. It is essentially trying to follow the gradient towards lower time-to-target values but would be constrained to available winds at available altitudes when making the choice of wind direction and altitude.

On an Arrival Time map, the value at each cell instead indicates how likely a balloon at that location will hit the target within the targeted time window; in other words, the likelihood of arrival time success. Again, the preferred direction can be found by calculating the likelihood-of-success gradient across the nearby cells, but, as always, the Steering Controller will be constrained by the reality of the actual winds, power available and other constraints.

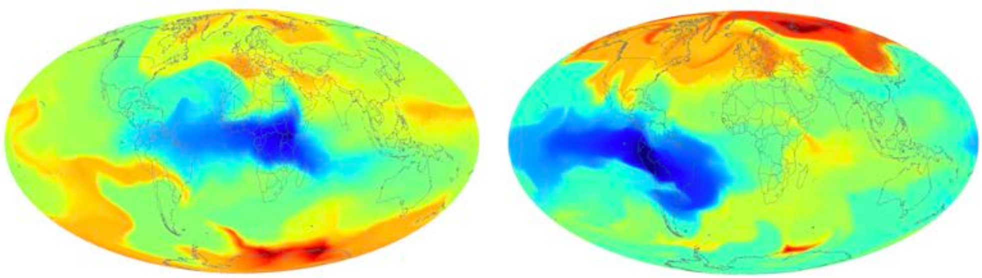


Figure 5-19 ASAP maps targeting Kenya (left) and Peru (right).

Cartographer computes new maps every time a new wind forecast is ingested. These maps are calculated in two major steps:

- Creation of a link map that lists all the cell-to-cell balloon motions that could occur, from each time slice (t) to the next (t+1), according to the forecast wind data.
- Calculating the time-to-target for every starting location, using backwards-propagating value iteration.

The link map describes, for each time t, and for every cell, all of the potential cells that can be reached from a balloon starting at that cell, for every 6-hour time step across the whole forecast window. For every cell at time step t, it determines and records all of the cells that can be reached by that balloon starting at that cell by step t+1 (6 hours later) and the likelihood of that happening.

The link maps are created with a series of independent simulations of a balloon drifting at a constant altitude for 6 hours for every starting cell and for each of a discrete set of altitudes. In this way, a step-by-step graph of all of the links between cells is formed.

The link map takes the form of a graph where cells are vertices and edges are a probabilistic estimate of transitions that may occur between cells. These transitions are the cell-to-cell balloon motions that could occur (the uncertain part) given the choice of altitudes (the decision within our control), according to the forecast wind data.

Then, starting at the farthest forecast point (10 days or 15 days) and building backward in time, Cartographer repeatedly calculates, for each cell, the fastest time-to-target for that cell, based on the time-to-target for the cells that can be reached and the time to get to those cells from this cell. In addition to the times, Cartographer also calculates the probability of arriving at the target, at any time.

At the first step (for the farthest future forecast value), Cartographer follows the links from the target cell backwards to cells that could reach the target in that forecast slice. All of those linked cells will then be updated to the value equal to the duration of the link between them and the target cell plus the value of the target cell itself, which is zero. The rest of the cells remain infinite.

On the second step, each of the cells has its predecessor links followed, again updating those predecessor cells with the value of the linked cell plus the duration of the link to the predecessor cells. This continues until the algorithm has backed all the way up to “now.”

***Figure 5-20** An illustration of Cartographer working backwards from the target.*

The evolution of wind currents can clearly be seen between these two snapshots which are generated from forecasts 24 hours apart. The blue regions represent a low number of days towards the target while the red regions are further away from the target.

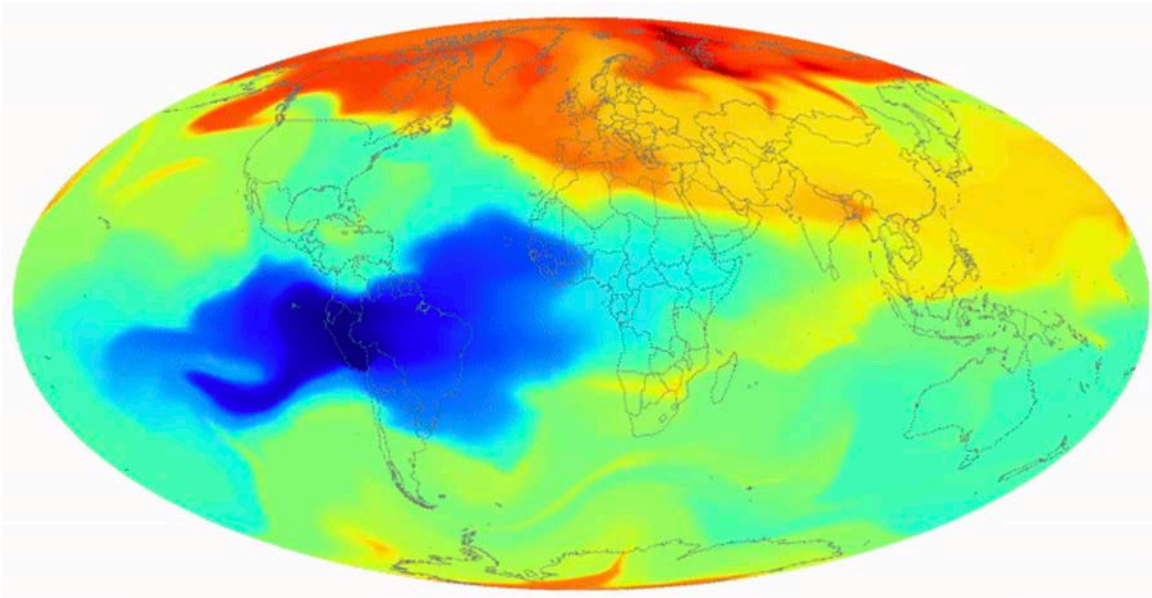


Figure 5-21 An ASAP map for Peru.

Notice the differences between Figure 5-21 (above) and Figure 5-22 (below).

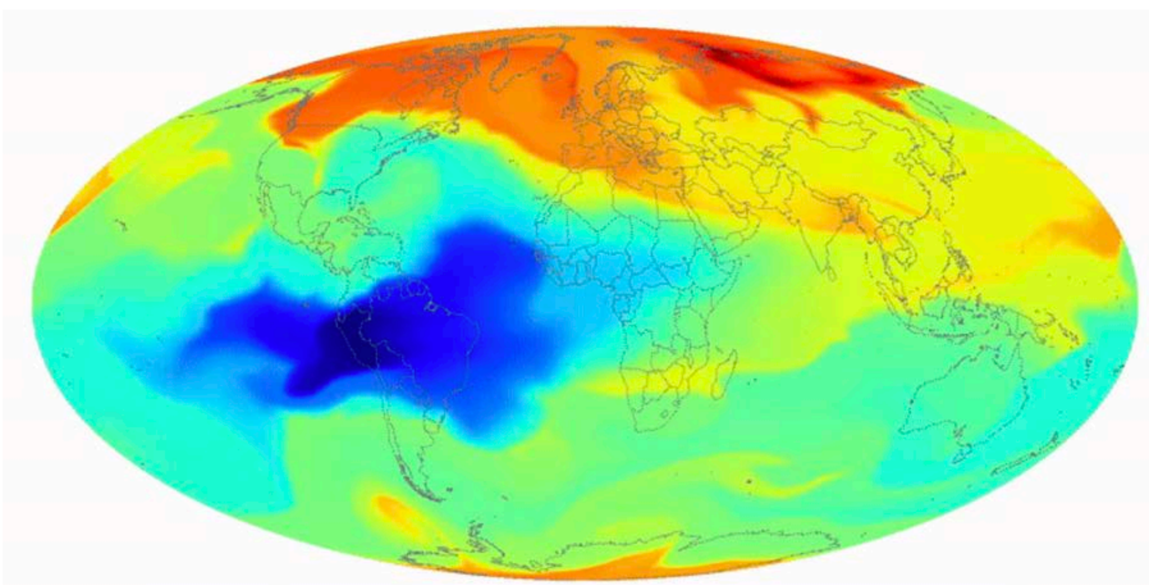


Figure 5-22 An ASAP map for the same target but 24 hours later.

For various reasons, such as geopolitics and military operations, some airspaces were considered "no-fly" zones for Loon. To automatically avoid such airspaces for balloons navigating long distances to their targets, Loon built the no-fly regions directly into the Cartographer maps by seeding these no-fly zones with a very high value (100 days) at the start of the backpropagation. This value would then spread out into directions where winds could potentially snare a balloon and force them into the no-fly. An example of a production map with integrated no-fly zones can be seen in Figure 5-23.

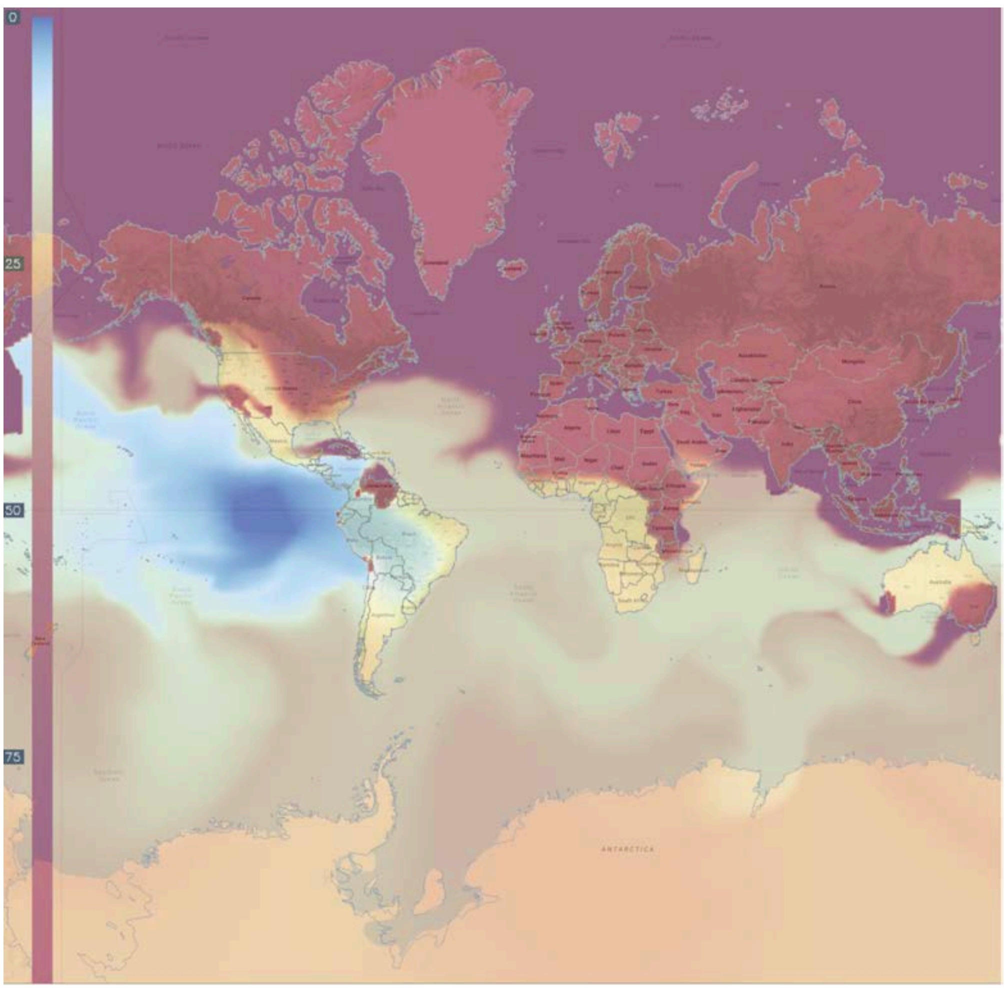


Figure 5-23 Production cartographer map with integrated no-fly zones (Cuba, Venezuela, Europe, Asia, North and East Africa, as well as smaller areas in Australia, USA, Peru or Colombia).
The deep red shows the bleeding of the initial no-fly values into directions of concern.

Cartographer Map Seeds

Cartographer maps are built using wind forecast data with new maps built after every new ingestion of wind data: every 12 hours. As soon as the processing of the data is finished and new maps are built (one to two hours after ingestion), they are used in production until the next forecast arrives so that Steering Controllers can use the most up-to-date information. However, the forecast from ECMWF was limited to a 10-day horizon, and NOAA GFS, 16-days, both of which are well below some common distant-target transit times. For instance, from Loon launch sites to Kenya at the start of commercial service, these transit times were often more than a month. Likewise, balloons that get blown away from South America are sometimes forced to circumnavigate the globe, again taking 20 to 30 days. The long-range navigation system then must accommodate these long transits that far exceed the horizon of the standard ECMWF HRES forecast.

Loon achieved this through seeding the cells on the map prior to the (backwards propagating) value iteration. These seeds indicated the typical amount of time to reach the target during historically similar periods that are good analogs to the current period. This analog could be as simple as an average across all historical data but this occasionally leads to values that could send the balloons off in a very wrong direction during “non-average” atmospheric flow periods. More reliable results can be attained by better matching the subset of the historical data such as matching the same time of year and other factors that help to match the current global state of the winds.

This seed calculation is based on the ECMWF ERA5 data from the past 30+ years. ERA5 (described earlier) is historical analysis data, estimating the winds that actually happened, during every 6 hour period over those many years. From this estimated wind data, a time series of time-to-target values across the extents of the ERA5 data-set for the globe can be calculated in a way very similar to the usual Cartographer link map and backwards-propagating value iteration calculations. Once calculated, summaries of the time series for each S2 grid cell can be computed in various ways to create the seed value: the initial time-to-target values for all cells, prior to performing the value iteration step.

Figure 5-24 shows, for the same target, a potential ASAP map that could be generated if a 2 month-long forecast was available, then if only a 0 day, 8 day or 15 day forecast is available. Calculating the historical average time-to-target for the full globe, gives seed values that create the final ASAP map shows in Figure 5-25.

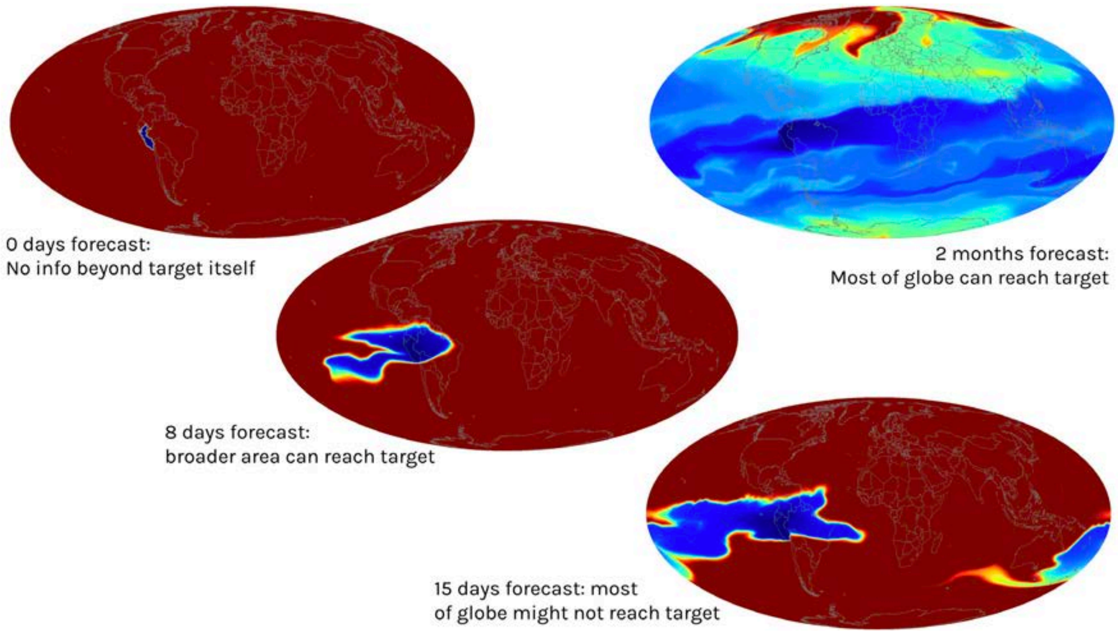


Figure 5-24 ***Reachability info for different forecast horizons.***
Within only 0 days, 8 days or even 15 days of forecast, most of the globe is considered not reachable. It's not until 2 months of reliable forecast (upper right) that routes to a target are available for most of the globe.

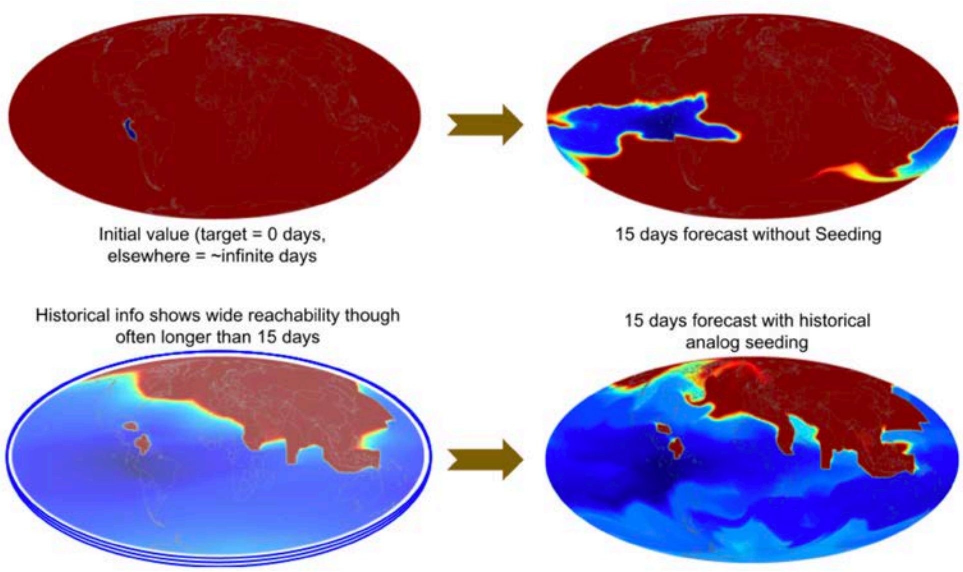


Figure 5-25 ***Seeding map building with historical-analog time-to-target values provides a map that can be used for very long-range navigation.***

As another example, look at a comparison for the Peru map with and without a seed, as shown in Figure 5-26 and Figure 5-27.

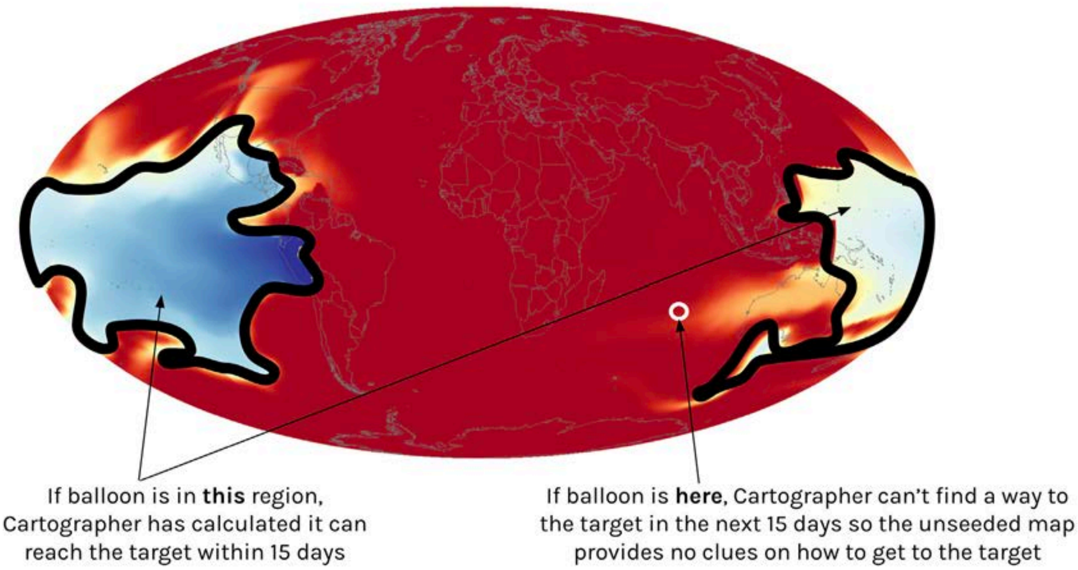


Figure 5-26 Peru unseeded map.

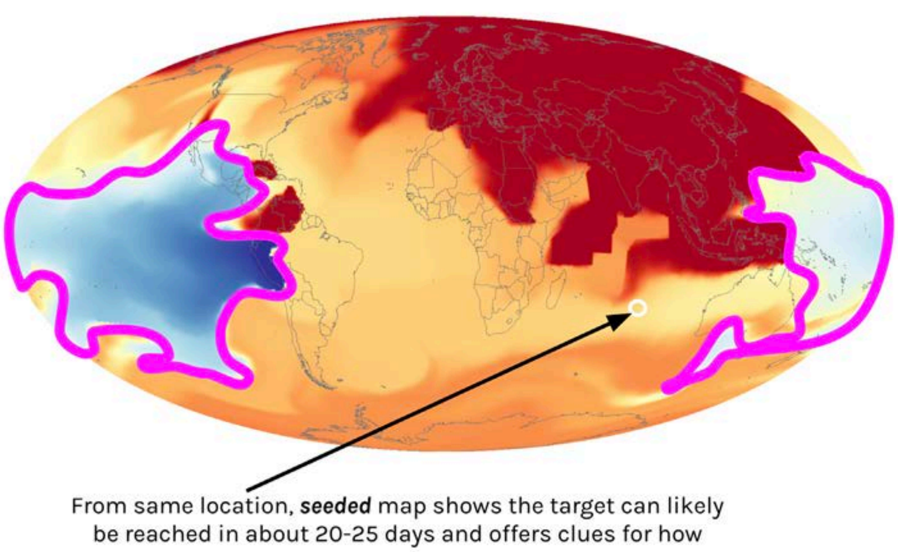


Figure 5-27 Peru seeded map.

Arrival Time Maps

As described in the Navigation Overview section, above, Dispatcher orchestrates a coordinated ballet of balloons flying in and out of a service region in a controlled fashion, which also avoids large clusters of balloons arriving at the same time, then all leaving at the same time. To accomplish this, Dispatcher assigns different balloons to different arrival time windows. Cartographer generates, for each target area, a series of Cartographer maps, one for each 12-hour arrival time over the 10-day forecast window. Dispatcher then will point each balloon's Steering Controller at the right Arrival Time map to maximize the likelihood that the balloon will arrive within the intended window.

Arrival Time maps are built using only forecast data; no seeds or historical data. The result is that we can only tell a balloon to arrive at a particular time within the next 10 days, the forecast horizon, but this is not a constraint since pre-scheduling arrivals that far out is already very low confidence.

Figure 5-28 shows an example of an Arrival Time map for central South America.

When reading an ASAP map, the values on the map are interpreted as the time-to-target, and the probability tells you the likelihood of making it to the target, ever. With an Arrival Time map, the values are instead the probability of making it to the target within the specified time window.

In these Arrival Time map images, we are looking at the probability of arriving in Southern Peru on 2019-11-19 00:00 UTC from six days earlier, 2019-11-13 00:00 UTC. Anything in the white region has a near 100% chance of arriving on time, while anything in the red region has 90% and so on. Anything in the black region has nearly 0% chance of arriving on time.

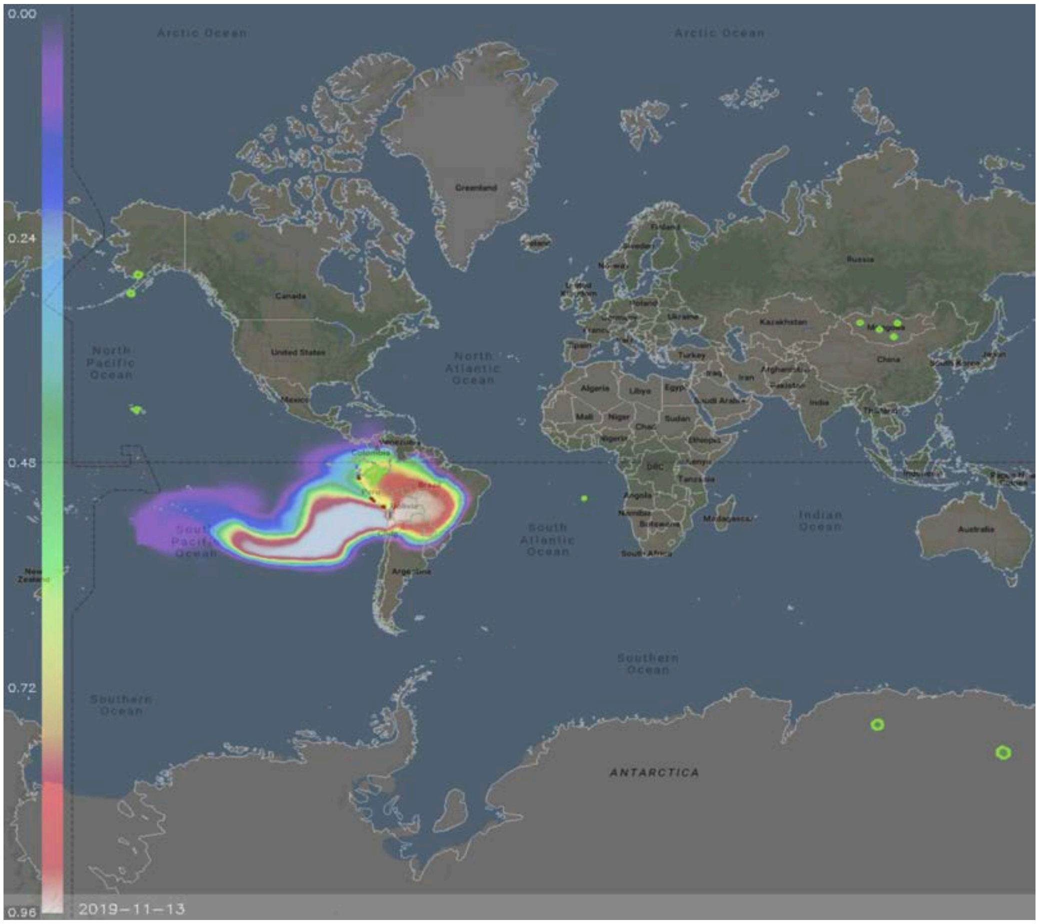


Figure 5-28 ***Peru Arrival Time map for arriving at 2019-11-19 (00:00 UTC) on 2019-11-13 (00:00 UTC).***

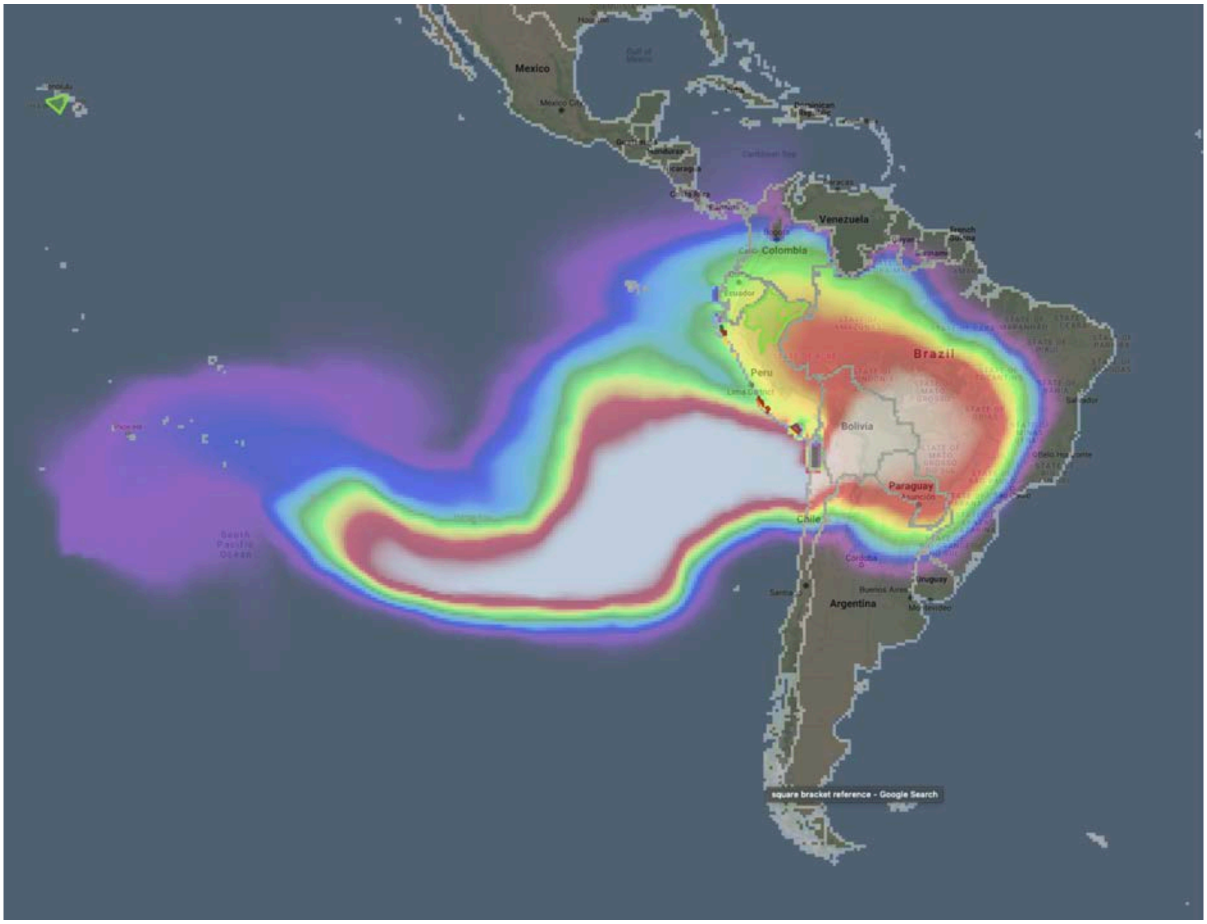


Figure 5-29 Peru Arrival Time map for arriving at 2019-11-19 from 6 days prior, 2019-11-13.

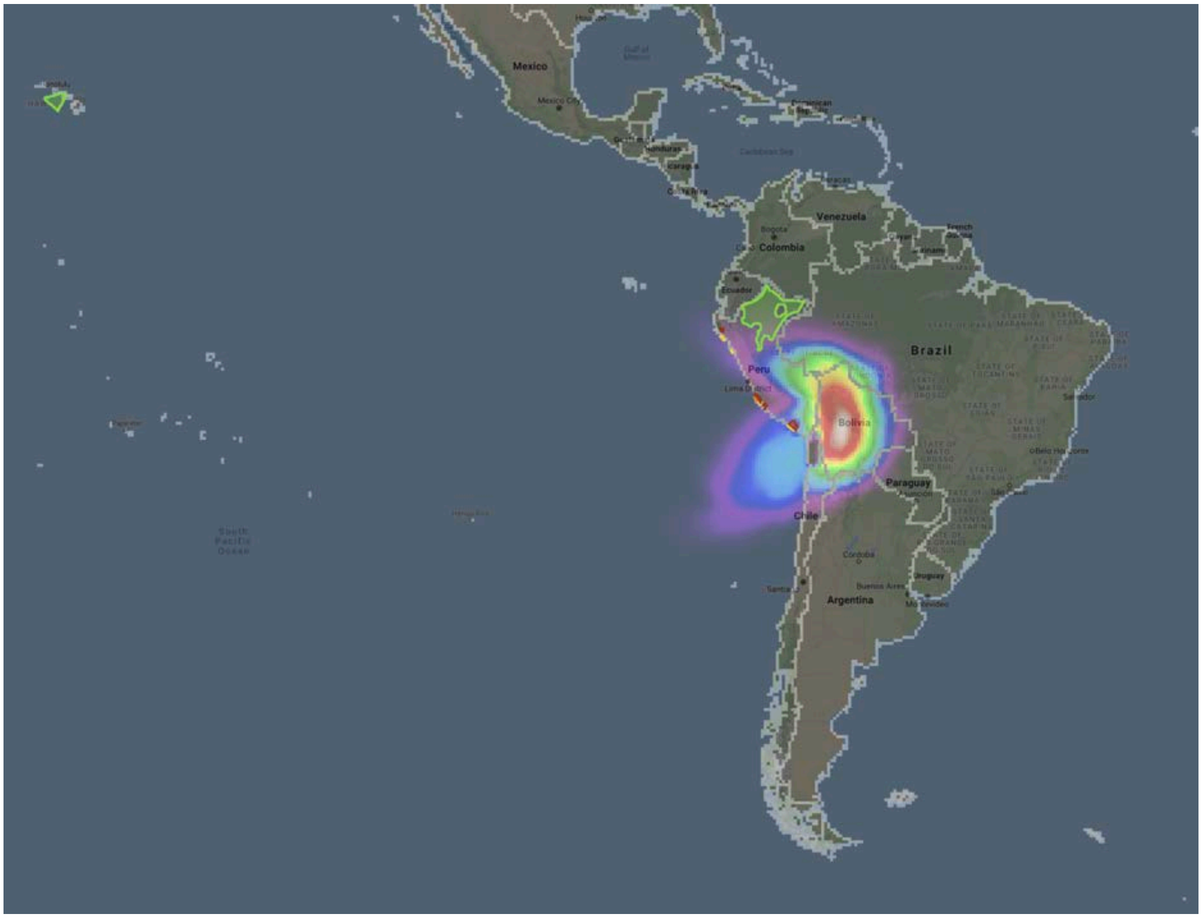


***Figure 5-30** Peru Arrival Time map for arriving at 2019-11-19 from two days prior, 2019-11-17.*

In Figure 5-28 and Figure 5-29, the target location and target time are the same, the only difference is the number of days away from the targeted window. The first map is 6 days from the target time, whereas the second map is 2 days away. Notice how the area where the balloon can be located to have more than a slim chance of success shrinks dramatically. It makes sense that as we get closer in time to our desired time, we need to be physically closer to our desired target.

Dispatch-Allocation of Flights to Dispatchers

The Dispatch system manages the fleet-level behavior of the balloons:

- As mentioned throughout the Steering Controller and Cartographer descriptions earlier, winds are often bad enough to blow any balloon that is in a service area far away, requiring that balloon to take days to weeks to make its way back to the service area. To maximize the availability of the service, a series of such blown-away balloons need to be arranged to float across the service area one after the other with minimal overlap and minimal gaps. Dispatch performs this orchestration.
- Balloon payload hardware may be configured to serve specific service areas and so may not be capable of serving others. The primary example of this was Loon's use of LTE band 28 in South America but LTE band 20 in Africa, requiring different hardware on those balloons. Vehicles configured for one were not useful for the other. Likewise, there are services, such as a backhaul service, which can be supported by either of the LTE configurations because it does not use the LTE system. But, even when the same balloon was able to support two different service areas, for instance, Mozambique and Kenya, both band 20, there may be business reasons (not hardware reasons) that a certain number or set of balloons are required for one service and a different set for another. Dispatch is involved with these types of orchestrations as well.
- There are different types of orchestration needed besides the mobile network expansion service that dominated the discussion. A Backhaul service for instance may need a very different set of coverage and timing rules. Likewise, some operations require didn't scheduling rules, such as managing all flights heading for a landing zone with a maximum number of landings per week.
- There are ephemeral conditions, such as an increased risk of no-fly incursion or a need to perform engineering tests, that warrant a short-term change of objective for a balloon. Dispatch is responsible for moving balloons from one objective to another, and, potentially, back again.
- And finally, Dispatch allowed the Flight Engineers to override its automation, to support special tests, demonstrations or other unique situations.

In all of these cases, Dispatch and the dispatchers that it utilizes, make use of the same locally-focused Steering Controllers and (usually) globally-oriented Cartographer maps to adjust the behavior of each balloon to meet Dispatch's overall fleet-level objectives.

At the top-level, Dispatch partitions the balloons into sets, called allocations, where each set corresponds typically to some business objective, e.g., LTE band 20 service in Africa, recovering old balloons, LTE band 28 service in Peru, or backhaul service in the Caribbean. This is done by a set of rules for hardware constraints and similar, or an auction for those flights that are capable of meeting multiple business objectives. The Smallworld UI allows Flight Engineers to modify existing rules but creation of new rules typically requires software changes. The auction splits the fleet based on relative priority of the business objectives at that time and can consider seasonal and regional demands as well as flight supply. Dispatch will repeat this process periodically, scanning the fleet and potentially re-allocating flights to different dispatchers as conditions change.

The auction process uses bidding across the dispatchers, with the highest bid winning. More involved intelligent bidding can be used with more elaborate rules factoring into the bids. An example of that would be bidding based on flight age, where a flight with the appropriate configuration would be bid upon by one or more of the service dispatchers, but after reaching its forecasted age limit, a recovery dispatcher would bid a higher amount on the flight to take control of it and manage its recovery.

Each of these allocations are then managed by a routine known as a dispatcher. For every flight allocated to a particular Dispatcher, that Dispatcher will assign a Steering Controller and an objective for the Steering Controller to follow.

In addition to this normal allocation of flights to Dispatchers, there is an additional Ephemeral Allocation that can temporarily override the primary Dispatcher the flight is assigned to.

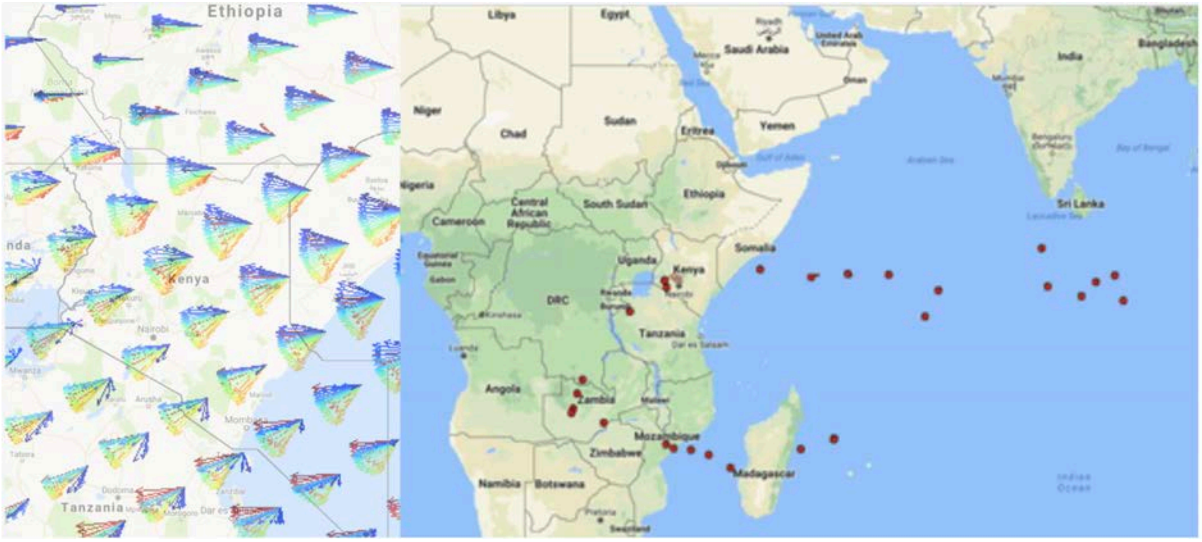


Figure 5-31 ***Oregon Dispatcher staging.***
An example simulation of fleet management through Oregon Dispatcher's staging and time-slot assigning to deliver continuous service across Keyna.

Oregon Dispatcher

Our service dispatcher, called Oregon Dispatcher, uses a task-based optimization scheme to match balloons to different time slots in the service region to maximize availability. Because Oregon Dispatcher is used for all of the MNE services across the world, multiple independent copies (or instances) of Oregon Dispatcher are active, each handling one allocation of balloons. As an example, one instance of Oregon Dispatcher could be assigned to all LTE band 20 service zones in Africa while another instance covers all LTE band 28 services in South America. A third could be assigned Backhaul services in the Caribbean. Each instance bids for the balloons that meet the requirements for its set of service areas and then manages the allocated balloons to maximize its objective.

Each instance of Oregon Dispatcher can be responsible for multiple discontiguous service zones but all balloons managed by that one instance will be capable of supporting any of those service zones. Each zone is a defined geographic area (defined in S2 cells) and contains one or more target locations. Flights allocated to the zone will be dispatched to the distributed targets with the overall objective to maximize availability and minimize availability gaps. Zones have several other parameters, such as a minimum and maximum number of balloons for that zone, and a defined circle (point and radius) within which to switch to a location-objective Steering Controller.

Oregon Dispatchers can also stash balloons in a stable holding pattern for more predictable delivery into the service area. The stash can be a time-based stash, to hold flights X days away from the target, or geographic, by specifying a predetermined area that has good maneuverability.

At a high level, it will assign flights to, for instance, arrive ASAP, arrive at a specific time, go to stashing zone (holds balloon some number of days away from the target).

Many different parameters could be tweaked, such as the frequency of planning, the horizon for which to plan, thresholds for no-fly and weather avoidance.

It is also possible to configure Oregon Dispatcher to have N flights arrive every X hours. If one needs to be even more specific, it is also possible to define which days of the week balloons should arrive at the target.

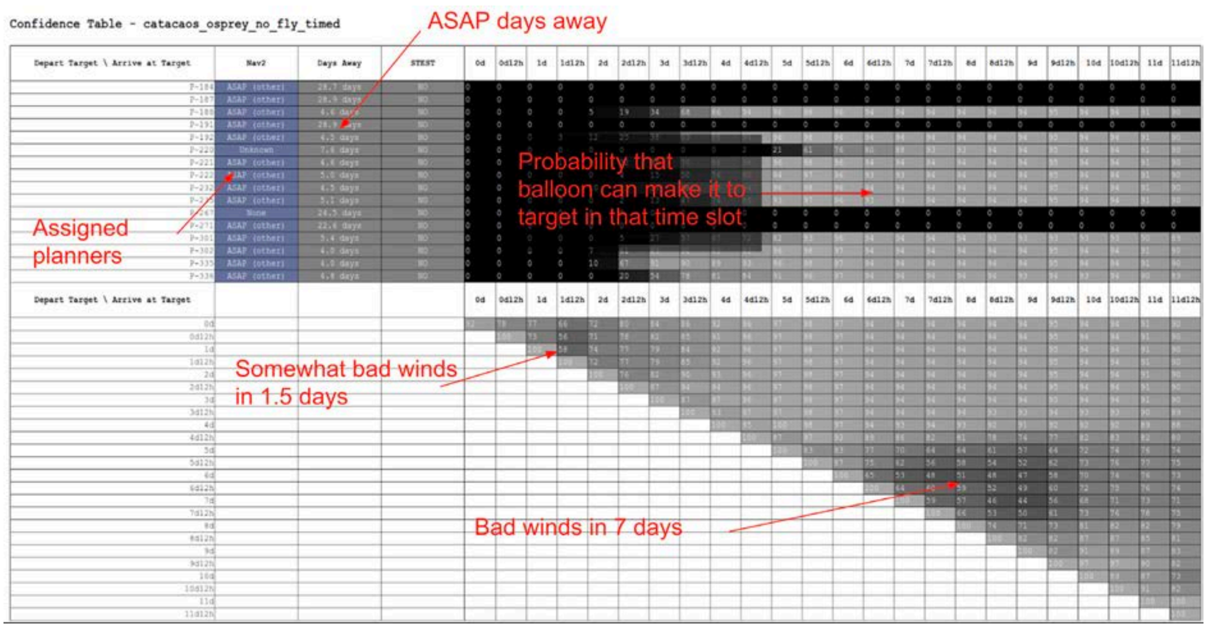


Figure 5-32 ***Oregon Dispatcher's Confidence Table.***
This provides flight engineers and others key information about upcoming service and confidence levels.

Nevada Dispatcher

Nevada Dispatcher is primarily used for end-of-life management of the fleet: managing pasture zones, landing and recovery tasking. It can also be used for other objectives such as long duration test flights or field tests. Nevada Dispatcher will bid for balloons nearing their end-of-life, and after allocation, direct those balloons to pasture zones where additional life testing can be performed or automatically select a landing/recovery zone for the flight.

Nevada Dispatchers incorporate similar features as Oregon Dispatchers, with the major difference being that Nevada Dispatchers are tasking balloons for individual and sometimes unique objectives, while Oregon Dispatchers orchestrate and schedule the group of balloons with the objective of providing as much service as possible.

Ephemeral Allocations

As previously mentioned, the Oregon Dispatcher optimizes the assignments of all balloons that are assigned to it. Occasionally an ephemeral event occurs that warrants taking temporary control of a balloon away from Oregon (or Nevada), while still leaving the assigned dispatcher responsible for managing the flight overall. This prevents the Oregon Dispatcher from reshuffling its entire set of flights, only to have to reshuffle them again after the ephemeral event passes.

A common example of their use is when a balloon, normally tasked by its Oregon Dispatcher, is unexpectedly heading for a no-fly zone. When this happens, Dispatch assigns a dispatcher to temporarily take control of the flight, steer it away from the no-fly, and return management back to Oregon after the balloon is back in a state that the assigned dispatcher can safely handle.

No-Fly Avoidance

As described above, all areas with flight or landing constraints are recorded and updated in an airspace details table, presentable on Smallworld and accessible to the navigation software. In the software, the no-fly regions fall into two categories: NORMAL and SEVERE. Examples of no-fly regions include countries for which we do not have general overflight permissions or named SUAs. Loon would avoid such regions either because they are forbidden or because they imposed a greater burden of air traffic coordination for our Flight Engineering team and the ATC team in a country. In either case, the objective was to avoid.

Loon defined no-fly zones with the appropriate priority over the no-fly region of the map. Given the reality of poor wind forecasts, defining those zones does not provide a guarantee that balloons will not traverse them.

Steering Controllers consider these no-fly zones and work to avoid them, but an even more effective tool is use of the long-range navigation maps. Cartographer, when generating its maps, uses an initial value of 100 days-to-target for all of the map cells within a no-fly zone, ensuring that the maps discourage routes that fly into these zones. In addition, Cartographer also explicitly builds an AwayNoFly map, with the explicit purpose of navigating balloons safely away from all no-fly zones.

The Oracle system provides regularly calculated estimates of the likelihood of entering no fly regions. Dispatch uses these percentage likelihoods to ephemerally allocate a balloon to the AwayNoFly controller, temporarily taking the balloon out of the control its primary Dispatcher. The AwayNoFly controller uses the AwayNoFly cartographer map to navigate the balloon to a location with < 10% likelihood of entering a no fly zone after 12 hrs.

Oregon Dispatcher only assigns balloons to a task if it predicts a certain probability of success. Similarly, it will not assign tasks that exceed a certain probability of entering a no-fly zone.

Loon had an excellent track record of avoiding such no-fly zones through the series of mechanisms just described: Steering Controllers actively avoiding those zones, Cartographer navigation maps discouraging risky routes near those zones, Dispatch ephemerally assigning AwayNoFly controllers when a vehicle is too close to a zone, and Oregon Dispatcher avoiding tasks with even marginal risk of entering a no-fly zone.

Storm Avoidance

As described in "The Stratosphere" on page 16, although there is almost no adverse weather where Loon operated, Loon's balloons did experience enough electrical activity and turbulence to cause concerns. We addressed those by making a number of improvements to the electronics and mechanical hardware design to harden the vehicle against the impact of these storms, if a balloon was subject to that situation. To further reduce the chances of damage, we also decided to actively avoid large storms, even if they were well below the balloon's altitude.

Loon's analysis showed that two values from the BCI/NCAR Global Weather Hazards project dataset were particularly well correlated with storms that could potentially cause trouble for Loon's balloons. These are the CDO (Convective Diagnostic Oceanic), shown in Figure 5-33, and CTH (Cloud Top Height), shown in Figure 5-34. In general, the larger the storm (correlated to CDO), the higher we wanted to be above it. This preferred altitude was referred to as the safe altitude floor.

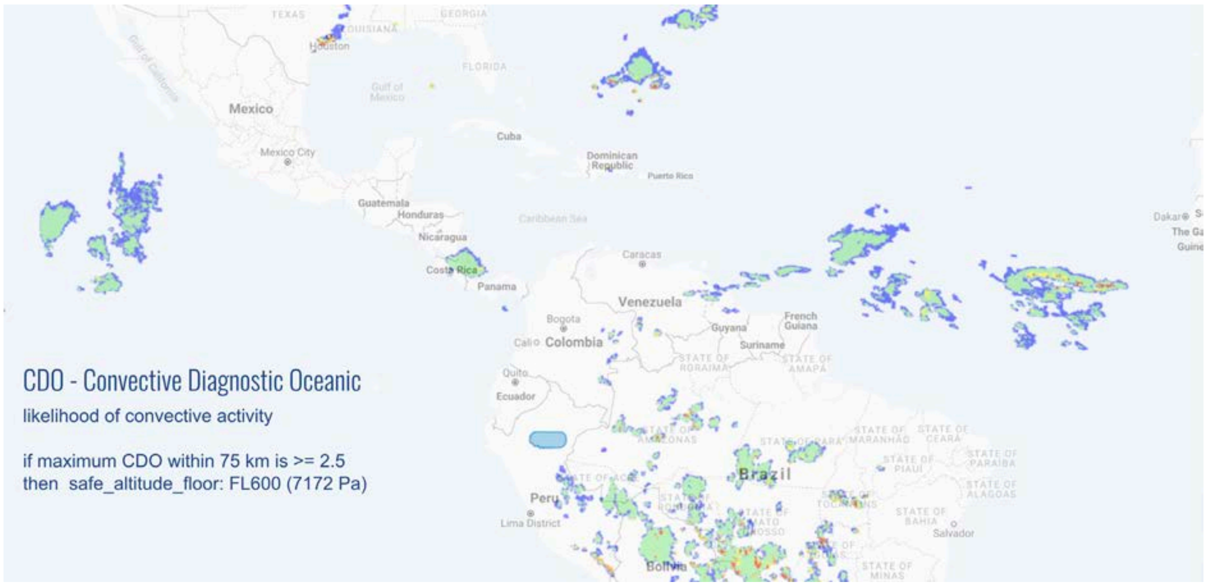


Figure 5-33 ***Convective Diagnostic Oceanic indicates likelihood of convective activity and, being generated by satellites, covers oceans as well as land.***

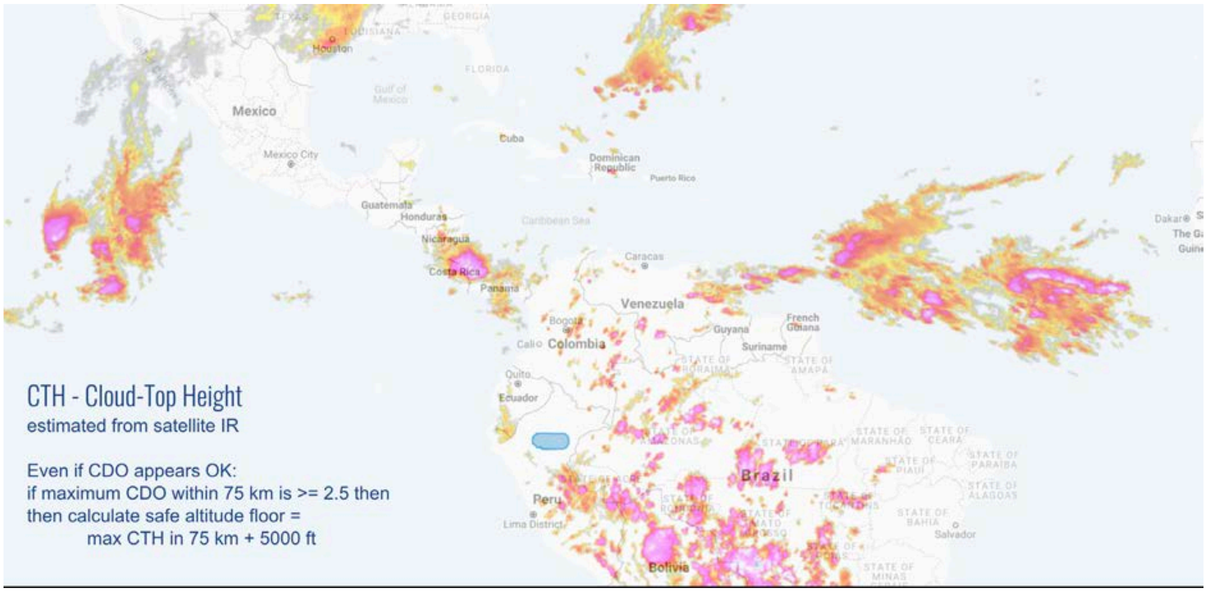


Figure 5-34 BCI/NCAR Cloud-Top Height was also used in calculation of Safe Altitude Floor.

To impose restrictions on the navigation system to incorporate this storm avoidance, we treated storms as dynamic altitude-restricted no-fly zones, with the initial parameters defined by the mentioned analysis and adopted rules of operations of the flight engineering team. The dynamic no-fly zones are created as circles above dangerous weather. For most of Loon's operations we used a smoothing radius (R) of 100 km. The safe altitude floor at a given point would either be 60,000 feet if CDO >= 2.5 anywhere within R, or the maximum cloud top height within R (in feet) + 5,000 feet (whichever was larger). The "safe altitude floor" layer was displayed in Smallworld as were the CDO and CTH values. Together, these provided flight engineers with good visibility into existing storms and storms that were forming and growing.

Figure 5-35 is an example of how the CDO and Cloud top height (CTH) layers are combined to generate the "safe altitude floor" dynamic no-fly zone map (right panel):

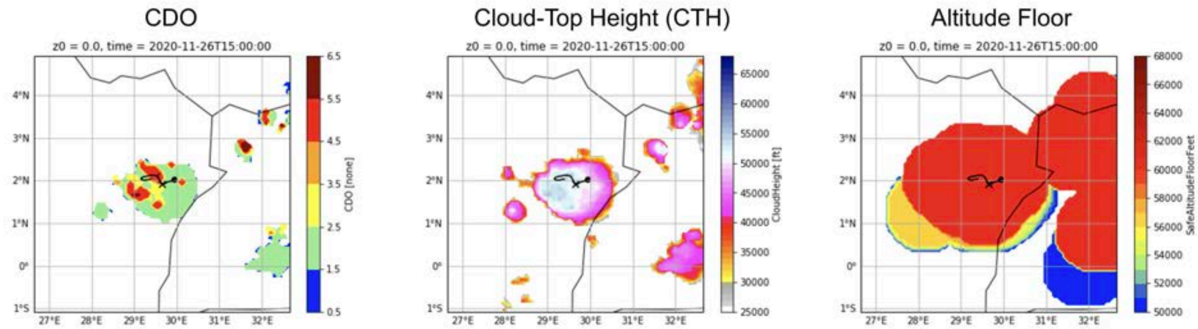


Figure 5-35 Storm-related input data types (CDO, CTH) and resulting calculated Safe Altitude Floor, created as no-fly zones for steering controllers to avoid.

Treating the stormy regions as dynamic no-fly zones was an elegant solution that allowed us to re-use all of the navigation system's logic built for avoiding no-fly zones to also avoid storms, both for classical and machine learning (Sleepwalk) Steering Controllers.

Most of our Steering Controllers actively sought to avoid intersecting a storm in their plan, minimizing risk of termination due to storm, either by ascending to the safe altitude floor or choosing winds that took the balloon away from the storm. However, in practice storms can develop rapidly and directly below the vehicle; in that case the action was simply to ascend. Powder ballast could be dropped manually by Flight Engineers to expedite the ascent.

Embedded Software

The Loon vehicles had 30+ embedded computers distributed across the balloon and payload and the ground stations had several. Loon built two different software platforms for these embedded systems:

- Major Tom: Designed for real time operation on a lower powered MCU used mostly within the Avionics subsystems
- Loonix: ChromeOS/Linux-based solution for the more complex applications within the communications systems

Major Tom–Avionics Systems

Most of the Avionics nodes on the vehicle used a smaller Cortex M4 processor (STM32F4). These systems also needed a real-time operating system, so Loon developed Major Tom, a platform based on FreeRTOS. Major Tom added significant enhancements to FreeRTOS to improve testing and test-ability, remote debuggability and remote monitoring to maximize the robustness of the avionics software. Loon, by policy, did not update avionics nodes after launch so extreme reliability and predictability were critical.

Some key features of these systems were:

- An intelligent telemetry, command, and control stack that operated over the SATCOM networks, as well as over the Loon backhaul network when present
- SD Card logging for storage of flight data and any error streams
- Onboard hardware fault monitoring with fault latching and acknowledging by ground-based fleet management system
- Processor crash scraping over SATCOM to provide register state and execution stack dump pre-crash
- A simulated flight environment with full system hardware to not only test each subsystem, but integration with communications and fleet management software
- LLVM sanitizers, static analysis, and custom compiler diagnostics to ensure firmware is further validated at compile-time
- Unit-testing and hardware-in-the-loop testing for each patch of code written to ensure a consistently reliable codebase

Loonix–Payload Communications Systems

For the communication systems, where a more complete operating system was required (and where there was a processor powerful enough to run it), Loon designed and built a variant of ChromeOS we called *Loonix*. The two primary objectives of Loonix were:

- Maximize developer efficiency through extensive re-use of many internal (proprietary) Google tools and libraries, including tight integration with a variety of Google data center services.
- Provide additional security features and containerization which made it easier to develop an inherently secure system to handle user data

Chapter 6

Engineering Production Software and Integrated Systems

Loon engineering for production software and integrated systems focused on four main areas:

Production Engineering/SRE

The hiring of Loon's first Site Reliability Engineer (SRE) in October of 2017 coincided with Loon's shift to deploying a full commercial service. The company had already validated its system through solid performance serving users after floods in Peru and after Hurricane Maria in Puerto Rico. It was time to transition to meet the rigorous and sustained demands of the mobile telecommunications market.

Beyond this commercial goal, there were also the increased demands of safely scaling the operation of all systems — a fleet of large, world roaming, super-pressure balloons, their ground stations, core networks and their command and control infrastructure. Building Loon's Production Engineering/SRE function on the foundation of Site Reliability Engineering would set up Loon to address both of these challenges well.

Approach

Site Reliability Engineering (SRE) is a software and systems engineering discipline developed at Google. See SRE Books[1]. Forged in the fires of web search and ads, it takes as its primary focus the application of software and systems engineering tools to building, managing and evolving highly reliable services that efficiently meet the business need. Core competencies include large scale/high performance service delivery, risk management, reliability assessment and implementations, scaling, monitoring and alerting, capacity planning, and incident response. In numerous cases, the Production Engineering/SRE team had to go back to first principles and modify approaches, solutions, and tools to address the novel problems that Loon presented.

Goals

Production Engineering/SRE at Loon had three primary goals:

1. Ensure that the fleet automation, management and safety critical systems were built and operated such that they met the high safety bar of the aviation industry
2. Lead the integration of the communications services (e.g., LTE) end-to-end
3. Own the mission of fielding and providing a reliable commercial service in the real world

Loon's objective was a commercial-grade mobile telecommunications (LTE) service from a (fundamentally R&D) stratospheric communications platform, attaining the levels of reliability and service expected of a terrestrial base station, but more cost effectively and over a much larger area, from a floating platform.

Design for Service Reliability

In addition to a special emphasis on designing systems for reliability, we implemented explicit graceful degradation features and systems at multiple levels. We rethought and adapted software releases to handle the intermittently connected, safety-related components that make up an autonomous system. We approached the novel debuggability problems presented by the system with creative engineering.

1 https://sre.google/books/

After several years of rapid iteration of dozens of hardware and software systems in parallel, we launched a focused end-to-end systems integration endeavor we called a Debug-a-Thon.

Graceful Degradation

Loon employed a layered approach to graceful degradation, supplying in-depth defense up and down the stack. It was most critical to prevent systemic failures (e.g., fleet wide). For example, an extended fleet-wide command and control outage could require Loon to ground the entire fleet. The combination of redundant systems, with the ability to quickly begin work on restoring a portion of the system that is down, is what positioned us well to address even the most severe problems.

Redundant Functionality Deployment

The autonomous systems needed to be redundant, modular, and decentralized to degrade gracefully. The capabilities of the software Loon deployed were meaningfully limited by deployment location and runtime resources. Critical functionality was implemented and deployed in a redundant manner to best balance system limitations.

For example, balloon navigation and higher-levels of power management were implemented in core data centers, with access to huge amounts of compute and storage resources (including ML accelerators). Loon could take in telemetry from all the balloons and use it to manage performance across the whole fleet.

If balloons were to lose access to the core data centers, however, they must still be able to function safely. As a result, Loon implemented a key set of navigation and power management functions in the embedded avionics software as well.

This is not to say that it is always preferable to implement logic in the data center. On the contrary, many functions — like reacting to a sudden loss of altitude — require quick action. In a case like this, software in the data center, which is a multi-minute round trip away, acts as the fallback for logic for code running on the embedded real time controller on the vehicle.

The goal was to balance optimal performance with redundancy to enable graceful degradation for all safety critical functions.

See the chapter: "Loon Software" on page 234.

Balloon Vehicle Automation

The autonomous balloon flight and management functions were implemented in the onboard avionics software and hardware. This included code to operate all equipment on board, and to maintain communications with data center systems.

To ensure the reliability of the critical telemetry and control channel, Loon provisioned two independent satcom paths: Iridium and Inmarsat. In addition, balloons could use the service backhaul network (B2x) for telemetry and control. Balloons continuously monitored these channels and adapted the amount and type of data to transfer, depending on current conditions.

When ground automation was unavailable (loss of connection or systems down), onboard systems performed the most critical basic tasks, including:

- Attain and maintain last instructed altitude.
- Ensure the balloon does not over-pressurize.
- Maintain last-instructed minimum battery charge-state profile.
- Begin descent after an amount of time (configurable by ground automation, e.g., three hours) over which no control/telemetry channel was functional.

The onboard systems also implemented the autonomous (i.e., no communication to the data center necessary) ability to navigate away from dense population centers before cutting down. See more on ground automation in "Loon Software Overview" on page 234 and on embedded avionics in "Avionics, Bus and Power Systems" on page 96.

Fleet Management System: Flight Vehicle Command and Control

Loon implemented Fleet Management Software functions as a constellation of data-center resident services and workloads. These functions included:

- Vehicle navigation and power management
- Wind prediction
- Vehicle simulation (both position and physics driven estimation models)
- Monitoring and alerting
- Rule-based automated control
- UIs for human operators

These systems were implemented as a distributed software system, running on Google's internal production infrastructure, spread across three data centers dispersed throughout the USA.

Physics-based estimators took in telemetry data as input into a model of balloon/payload state and conditions. The model drove fleet and balloon commands (e.g., ascend, descend, and point towards the sun). Loon provided fallbacks to estimators that couldn't run due to lack of data. See more in "Fleet Management Systems" on page 235.

Edge and Field Deployments

Loon deployed systems at the edge — typically in its partner MNO's data centers — to provide for high performance and high reliability interconnection with MNO core telecommunications networks. They also served to partition the failure domain for these systems; edge and field deployment failures could, at most, affect only a single service country. Loon was able to employ tried-and-true, industry standard methods for provisioning redundant systems in these environments (e.g., using HA pairs, load balancing, and multi-datacenter deployments).

Loon also deployed ground stations in the field, close to our LTE service regions. These installations consisted of:

- Tracking, directional, E-Band antennas that communicated with balloons
- A set of networking equipment that enabled the sites to connect to the internet

Loon used enterprise internet connections at these installations to provide backhaul from these sites to our edge equipment.

The choice of deployment sites for ground stations was considered carefully. Loon needed to find places that had access to reliable network and power, and that was also within range of the balloon-to-ground radio frequency connections, up to approximately 130 km. Loon also sought out locations that had uncorrelated storm activity. Because rain is opaque to E-Band, Loon used the number and geographical dispersion of ground stations to reduce the likelihood that all balloon-to-ground connections would be inaccessible simultaneously.

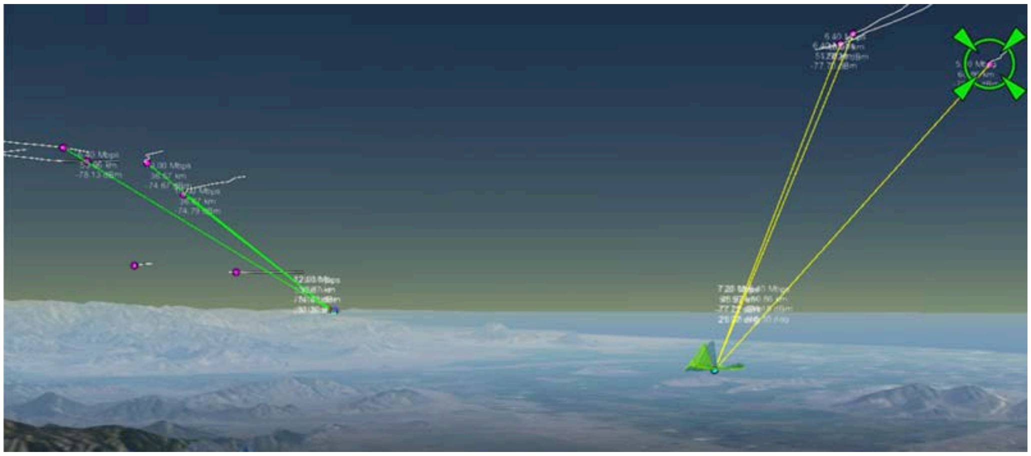


Figure 6-1 Balloon-to-ground links from geographically separated ground station sites.

TS-SDN

Loon built and deployed a temporospatial software defined network (TS-SDN[2]) to orchestrate the automatic construction of an ad hoc network between stratospheric and ground systems. It exploited the multiple E-Band antennas on each payload and at each ground station site to enable it to instantiate reliable services by dynamically adjusting to constantly varying operating conditions (e.g., movement due to wind driven navigation, obstructions, or localized atmospheric conditions). TS-SDN commanded redundancy-enabled graceful degradation of the service network in the face of link failures and unexpected changes.

2 Temporospatial Software Defined Network. B. Barritt, T. Kichkaylo, K. Mandke, A. Zalcman and V. Lin, "Operating a UAV mesh & internet backhaul network using temporospatial SDN," 2017 IEEE Aerospace Conference, 2017, pp. 1-7, doi: 10.1109/AERO.2017.7943701.

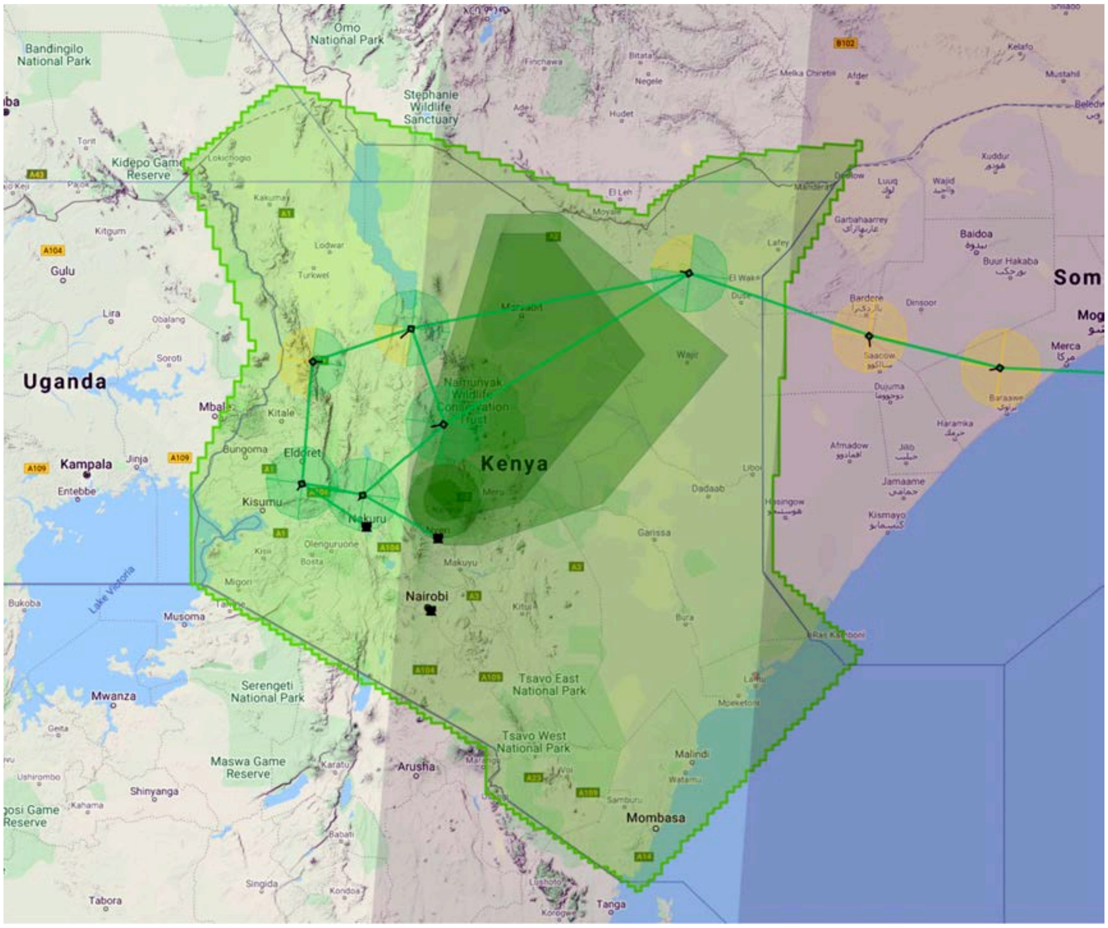


Figure 6-2 TS-SDN commanded ad-hoc service mesh over Kenya at dusk.

Key Learnings

Loon gained the following essential understanding:

- Layered approach to graceful degradation was effective in limiting the risk of catastrophic incidents
- Mesh network with multiple ground stations provided good service resilience

Challenges

Loon faced the following challenges:

- Network robustness was not a first-class objective of the Minkowski solver, and it was becoming increasingly clear that it would need to be.
- System was not resilient to fiber/backhaul cuts between ground stations and the MNO core networks, yielding meaningful (yet still rare) service outages.
- Global flight vehicle and communications control plane was implemented as a global failure domain. Partitioning would clearly be needed in the future.

Software Releases for Autonomous Systems

In addition to applying Production Engineering/SRE principles to deploying software updates in an unusual environment, a novel challenge at Loon was doing this in the context of running a safety critical autonomous airborne fleet.

Safety Related Functions

Safety-critical software functions were distributed between the onboard avionics and various data-center jobs, in a way that leveraged each computing platform's strength and ensured a graceful degradation. For example, altitude control functions were performed on-board but configured via ground automation instructions. Similarly, power consumption limits were implemented on board but dynamically configured by ground automation. This approach enabled the leverage of data center power to perform sophisticated safety-related functions (e.g., storm avoidance) while ensuring that the vehicle could continue to perform essential functions in case of loss of communication to the Fleet Management System. As a result, Loon needed to manage software releases in a manner that was attendant to the potential safety risks of components across all of its runtime environments. Loon accomplished this using an accelerated release cadence.

Push on Green

As Loon built and deployed systems, it set out to enable push-on-green[3] capabilities up and down the stack. A push-on-green release model for deploying production software employs a strategy in which every version of the software that passes all

3 D. V. Klein, D. M. Betser, and M. G. Monroe, "Making Push On Green a Reality", in ;login:, vol. 39, no. 5, October 2014.

the tests is deployed. There are many advantages to this approach, perhaps the most important of which is that each release that goes out is small (i.e., contains a small number of changes/features, or a limited amount of changed code).

Experience shows that pushing a larger number of smaller releases is less risky than a smaller number of large releases, easier to identify the cause of a breakage, and easier to roll back to a known good release (where rollback is possible.)

What Loon found was that while push-on-green was within reach for its datacenter workloads, the team had to adapt the release model significantly to accommodate the constraints of the operating environment. These adaptations included:

- Implementing a tiered release cadence that was sensitive to the safety profile of components
- Accommodating more stringent interoperability testing requirements
- Adjusting to the limitations of intermittent connectivity

Release Cadence

Release frequency goals were informed by the risk, potential cost of failure, and difficulty of recovery of the system's various components. While many of the datacenter systems were global in scope, they were also easiest to recover in the case of a bad software push.

Towards the periphery of Loon's deployed systems, bad releases had the potential to take out a smaller and smaller percentage of aggregate capacity (down to a single balloon or a payload subsystem) but were also more difficult to recover. At the extreme, the avionics software and firmware, which controlled the hardware onboard to actually fly the balloon, was deemed so critical that Loon never updated it after launch.

The release cadence goals were organized as follows:

- Datacenter: Push code on demand (no practical limit; at least weekly, with support for limited live configuration updates that allowed modifications between releases)
- Edge (partner's data center): Weekly target due to tight integration with telco partner network and resources
- Comms payload: Weekly target for embedded comms systems, gated on integration testing (more like four-to-six-week cadence in practice)
- Avionics: never updated in flight (bugs were actively reviewed, and could have been patched in-flight if risk was high enough)

Loon believed that this practice of continuous improvement was one of the key contributing factors to its overall (exceptional) safety record, converting punctuated anxiety about big, complicated changes into a regular cadence of bite-sized updates. Supporting this higher rate of change required Loon to invest significantly in release integration testing.

Validation and Testing

The layered validation, tests and test environments consisted of:

- Code reviews required for all submissions, both software and configuration, and all changes tracked in formal source control systems. No exceptions!
- Pre-submit unit tests: Code functional testing that gated submission.
- Post-submit tests: Unit and integration testing at the subsystem level that occurred after each code submission.
- Comms tray testing: Hardware in the loop (HIL) testing of comms equipment and functionality.
- TROOM: HIL integration testing including avionics and comms at room temperatures.TRON: HIL integration testing including avionics and comms at stratospheric temperatures (below -60 °C/213 K…COLD!)
- WMC Testbed: Four-node (two ground stations, two payload simulators) physical testbed set up in Winnemucca, Nevada that ran a pre-release version of the software end-to-end, testing network functionality including all hardware and data center components.

Canarying

After a release passed through the integration test environments, Loon then undertook a staged rollout to the fleet aloft. The team selected a representative node to update and pushed the new code to it. The code was allowed to run, usually for a day or so, before pushing to progressively larger numbers of nodes. This approach, a best practice in the data center, proved useful, if harder to manage, in the mobile fleet. Sometimes, for example, a balloon selected to be a canary would float out of the service area and not reconnect for a substantial amount of time. The team would have to select another canary and restart the process when this occurred.

The process of selecting and validating canaries is eminently automatable, and Loon was in the middle of doing so when it shut down.

Safety Impact

Loon strongly believed that rapidly iterating and moving fast was worth the cost of degrading some comms equipment in the stratosphere. Loon also believed that rapid and small software iterations were essential to ensure the safety of operations which could not be compromised on "Aviation Safety" on page 347. No testing complex will catch all bugs, and some bugs will brick a system (i.e., render it unresponsive and unrecoverable). Given the cost and number of vehicles, it was reasonable to trade rapid iteration for isolated instances of degraded comms equipment. Loon never settled on a specific target but did end up rendering two comms payloads unusable (out of hundreds flown) in the course of bringing up and operating the service. By design, the avionics systems on these balloons were unaffected, and they remained fully controllable, monitorable, and landed normally and safely.

Key Learnings

Loon gained the following essential understanding:

- Frequent software release is important for safety in a rapid development environment – it enables effective continuous improvement of ground automation.
- Automated testing scales to catch problems in a system with many interdependencies.
- Testing with Hardware in the Loop is vital to ensuring that releases are safe and behave as expected.
- Ability to implement problem mitigations quickly (e.g., on-ground automation with dynamic configuration changes) was important for safety.
- The improvements Loon obtained by moving faster mollified those initially resistant to the idea that the occasional loss of a comms nodes was acceptable.

Challenges

Loon faced the following challenges:

- Integration testing on physical hardware was the most significant impediment and bottleneck in the system.
- Dynamic configuration updates without proper canarying were a ticking time bomb (one Loon knew about and, thankfully, never went off).

Service Debuggability and Debugging

Loon had been building and operating a fleet of stratospheric balloons for more than five years when the market and business priorities aligned behind entering commercial service. This meant that the vehicle and fleet management tools were quite mature and feature-rich and they provided a good foundation upon which to build a service. Vehicle and fleet management was implemented as a constellation of systems and services and presented in a unified UI called *Smallworld.*

Novel Challenges

Debugging Loon's communication platform services, it turns out, presented a novel set of challenges relative to deploying and managing services from data centers or the edge. These challenges included:

- Incorporating temporospatial information into all debugging flows
- Lack of control over constructing repeatable conditions
- Extreme bandwidth limitations of the satcom command and control channel
- Intermittent connectivity of nodes
- Importance of supporting time travel in debugging/visualization tools
- Difficulty moving up the hierarchy of explainability

Temporospatial Awareness

Given the time and space dynamic nature of the stratospheric platform Loon was operating, capturing the physical and geographic state of the system was virtually always the first step in any service-level debugging. Aspects such as balloon position, payload orientation, storm activity, and population density are critical factors to service delivery, and are all able (likely!) to change while debugging. Responding to an alert that fired five minutes ago may find the investigating engineer examining a system in a completely different state than the one that alerted.

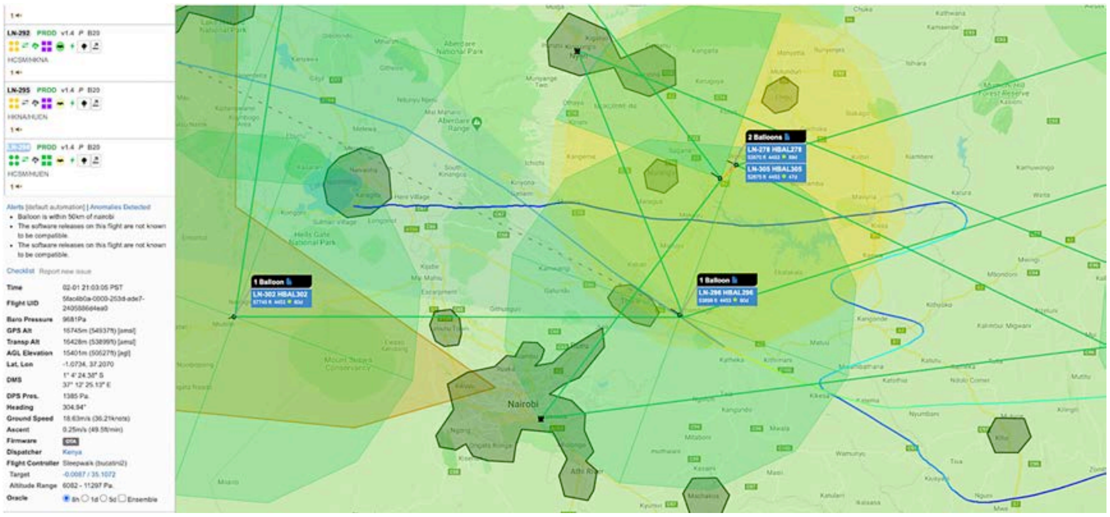


Figure 6-3 Snapshot of LN-296 position captured at the beginning of a debugging session.

Repeatability

The inability to consistently construct repeatable conditions for debugging also substantially added to the difficulty of tracking problems to root causes. In the early days of bringing up the balloon-to-ground E-Band, point-to-point network links, the team observed conditions that should have resulted in closing the connection but did not do so. When the balloon passed out of range, Loon could not simply send it around again to try again. Depending on the wind, that balloon could be back in 3-5 days, and could return on the opposite side of the ground station (i.e., use different antennas to attempt the link).

High Latency/Low Data-Rate

Having no physical access to systems in the field presented the challenges an SRE or SWE might expect if they have experience debugging remote data center systems (i.e., with no physical access). A significant difference for Loon's systems, though, was the extreme bandwidth limitations of the satcom debugging channel. This was the very same satcom channel used for command and telemetry. This channel was deployed using Iridium's Short Burst Data (SBD) (and eventually Inmarsat's IsatData Pro, IDP) service, which supports a message payload of 1-3 KB of data with a median roundtrip of 1-3 minutes. It would not be wrong to think about the challenge as similar to debugging servers in a datacenter using SMS messages alone.

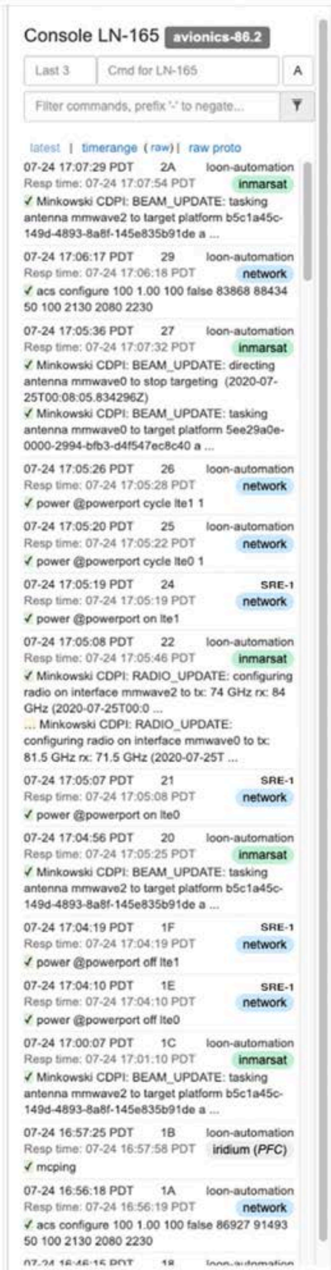


Figure 6-4 ***Smallworld command console with time stamped messages and transit path.***

This extreme constraint led to the implementation of a complicated data packing scheme to enable command, telemetry, and debugging over the low data-rate channel. Vehicle and payload properties and characteristics were compressed and placed into priority bins: primary, secondary and best effort. Variables in the primary bucket — things like position, altitude, and velocity — were packed into every message sent back from the payloads. Others were accommodated based on space and priority. These variables fed estimators, simple models implemented in software in the data center that fed information into the vehicle and fleet management systems. It was extremely easy to overwhelm the satcom channel, even before adding debugging information onto it.

Intermittent Connectivity

The fact that Loon was deploying a high bandwidth/low latency service like LTE with the platform provided another opportunity (and challenge) for debugging, and that was for in-band communications over an E-Band backhaul connection. For the relatively small fraction of a balloon's life that it spent connected to the service mesh, Loon was able to get high-rate telemetry (HRT) and use tools like ssh to log in to payloads to debug systems. This enabled a huge step change in visibility and control, critical to debug the orchestration of Loon's TS-SDN system and the complexities of LTE.

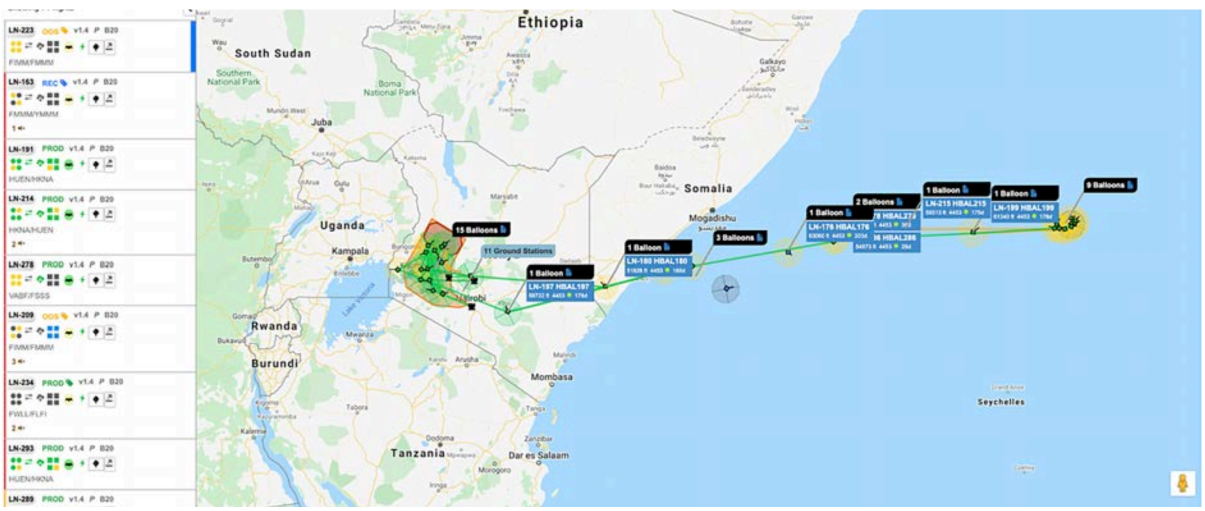


Figure 6-5 ***Intermittent connectivity.***
A perfect mesh for updating software, but with a single isolated balloon, unable to connect for lack of sufficient power.

Along with opportunity, using in-band communications presented another challenge, which is working with nodes that have intermittent high bandwidth connections to Loon's core systems. In general, the debuggability of subsystems was reduced as one went from core to edge:

- Data centers: good visibility, mature tools and systems
- Edge: still good visibility and access, mature tools and systems
- Floating/intermittent connectivity: much harder to deal with

Intermittent connectivity led Loon to try to reuse tools that were designed for data centers, with an assumption of uninterrupted network access, for the payloads with their inconsistent access. These tools included the core internal monitoring and telemetry gathering systems. Loon wrote a component called Steward whose job it was to bridge telemetry coming from intermittently connected payloads to Google's monitoring systems but struggled with the mismatch of requirements. Loon had repeated problems with data consistency, where payload data was not written to all monitoring replicas and so, were forced to add complexity to monitoring rules (where they were unwanted) to work around it.

Time Travel

A key lesson for Loon was when it learned the value of supporting "time travel" in the UIs and debugging interfaces. To understand the state of the system at a given time, it was very often necessary to understand its prior state in a physical and geographical context. Smallworld had support for visualizing the historical path of the balloons and Loon added similar capabilities to Minkowski's NetOps UI, the UI for managing mission concerns. The NetOps UI presented information about the various layers of the network — physical <-> control <-> data <-> overlay (LTE) — and gave it the capability to show this multi-level status at arbitrary points in time in the past. Loon found that it was critical to present both the predicted and actual state at these historical points in time. Once it built these features into the debugging UI, Loon found that cognitive load for on-call engineers dropped significantly.

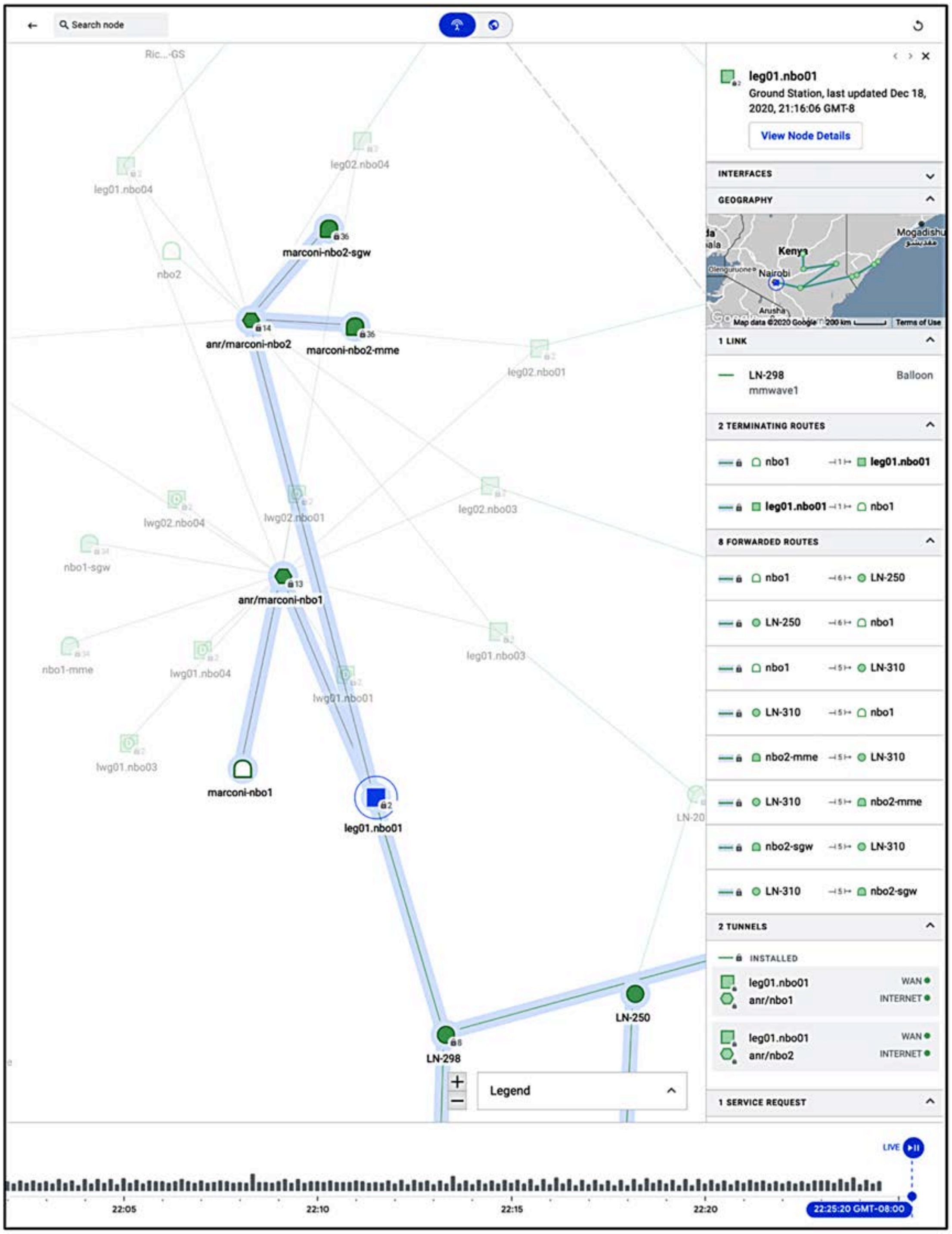


Figure 6-6 ***Network topology centered on the nbo01 ground station in the NetOps UI, with time travel scrubber visible at the bottom of the interface.***

Explainability

An interesting and pernicious problem that persisted even in the face of the NetOps UI was that of the hierarchy of explainability. When Loon first started integrating the system, we often had trouble understanding what was happening across the network, what was affecting what and why. Over time, Loon built tools increasing visibility into these relationships making it easier to understand the broad current status. Adding the time dimension enabled Loon to understand how it got into the current state.

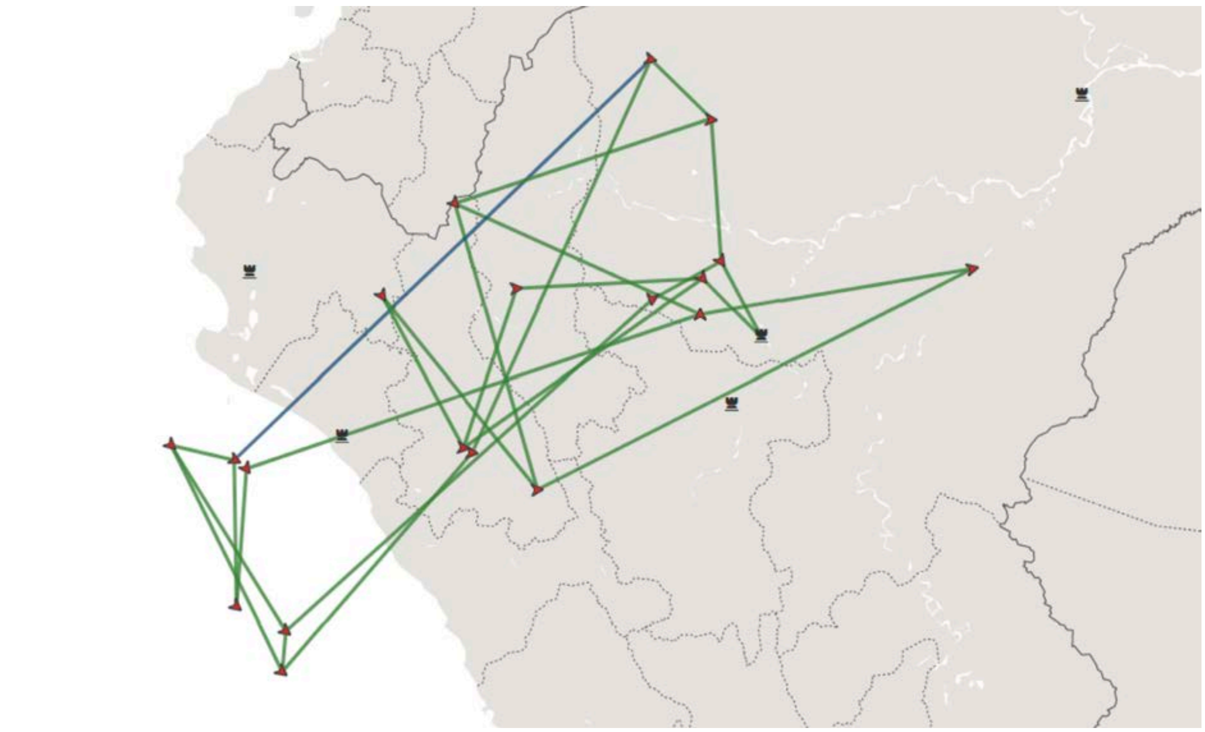


Figure 6-7 ***Mesh with single point of failure.***
A sizable mesh of 19 balloons over Peru, connected to a single ground station site (single failure domain) despite there being other ground station sites (indicated as rooks) available and within range.

The hierarchy of understanding encompassed three questions:

1. *What is the current state?*
2. *How did we get into this state? (What were the previous steps and states?)*
3. *Why are we in this state and in no other states?*

Explainability is a big topic of discussion in ML systems. It is notable that a lack of explainability presented a significant stumbling block for understanding the choices made by the non-ML Minkowski solver as well. This system creates the backhaul network topology and routes but Loon had not yet created a strong visualization of why the system had chosen a particular target state over other potential options. Additional future work was aimed at making network robustness a direct and visible objective of the solver (along with requested connectivity).

Minkowski Drains

Bringing a node offline for inspection or to mitigate a service outage is a key capability for managing highly available systems. To do this on the Minkowski network, Loon implemented administrative *drains*. This presented particular complexities because of the fact that nodes were a part of a mesh network. Loon implemented both opportunistic and immediate drains. Opportunistic drains would wait until a node was not being used for user traffic, and then put a block in place preventing new user traffic from being added to the node. Immediate drains were enacted immediately and would cause any user traffic traversing the node to be rerouted. In the case that no other viable route was available or the rerouting was slow, it would cause the loss of that user traffic. Loon had plans to improve this feature's ease of use and safety of use.

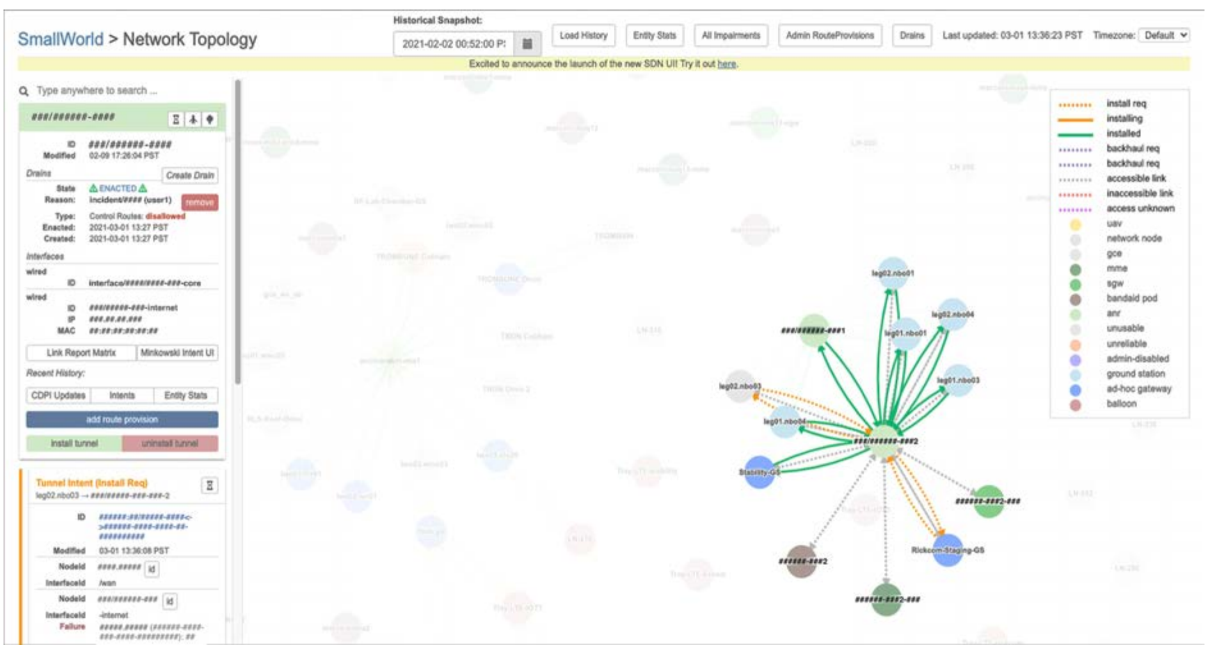


Figure 6-8 ***Minkowski drain control UI.***

Pets vs. Cattle Dashboards

Loon used two separate dashboarding systems for debugging, one optimized for pets and the other for cattle.[4]

The first was purpose-built into Smallworld. It made it very easy to visualize graphs of a single balloon's payload and physical metrics. It was a very useful tool for digging down into a particular problem or a specific vehicle. Smallworld allowed users to very easily compose dashboards on the fly to assess many different aspects of a vehicle that were relevant to a particular problem. It was a tool optimized for monitoring pets.

Figure 6-9 ***Radio frequency telemetry in Smallworld, showing multiple aspects of signal strength and quality for an antenna on a single balloon.***

The second dashboarding system that Loon used was a standard Google internal system. It provided communications processes and properties centric monitoring. It was good at aggregate and cross-sectional views of the service. Loon bolted on support for slicing by the dynamic geographic nature of the service (i.e., machines in data centers don't move around very quickly). This is a tool optimized for monitoring cattle.

4 http://cloudscaling.com/blog/cloud-computing/the-history-of-pets-vs-cattle/

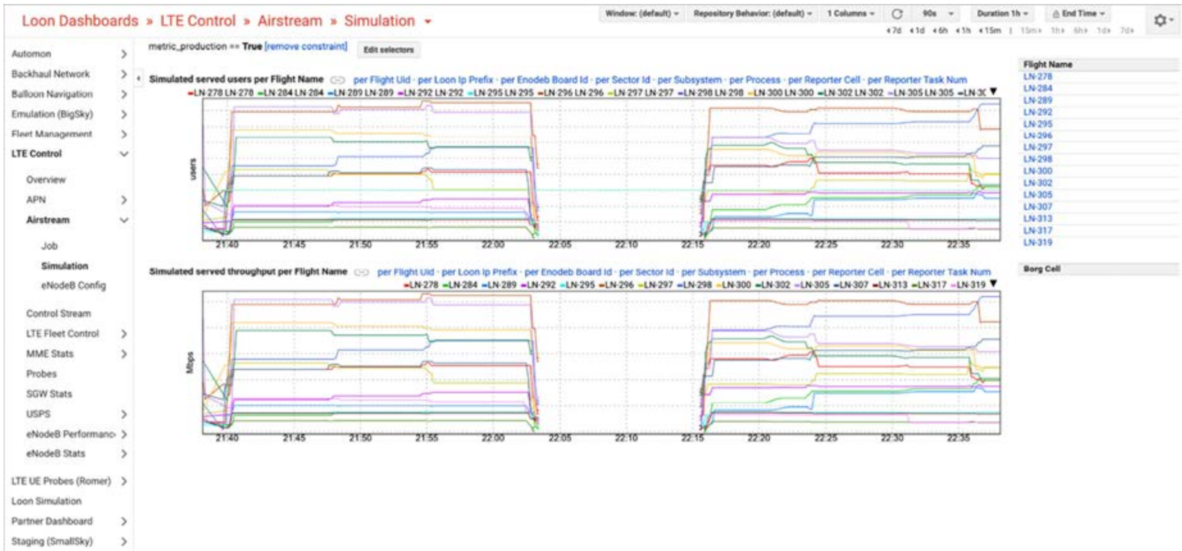


Figure 6-10 ***Dashboard showing radio frequency simulation results for Loon's fleet of balloons over Kenya.***

Each of these systems played an important role in Loon's ability to deploy a fleet of autonomous systems and manage a temporospatial LTE network. The question remains, though: Did Loon need both? One way of evaluating this is to ask what it would take to implement the core features of one in terms of the other. With some work, it would be possible to use just one of these tools for dashboarding, but it would be reducing complexity at the cost of speed of execution.

Key Learnings

Loon gained the following essential understanding:

- There was a physical and geographical aspect to all debugging.
- Being able to look back in time is critical. Must store both predicted and actual state, and present those to get a good picture.
- Use of low data-rate SATCOM service was workable, but significantly slowed the speed of integration, deployment, and debugging.
- Debugging support for different levels of granularity (both pets and cattle) was critical for success.
- Well-conceived and built UIs dramatically reduced cognitive load.

Challenges

Loon faced the following challenges:

- Decision to limit telemetry and control channels to low data-rate interfaces unable to support SSH was a premature optimization, made at a time when the expected number of balloons was far higher.
- It would have been good to take inspiration from MPLS/SRv6 monitoring in industry.
- Using a mixture of quickly-built, single-purpose systems and off-the-shelf solutions for the debugging and monitoring dashboards saved Loon development time but added debugging complexity.

End-to-End Service Integration: Debug-a-Thon

The Production Engineering/SRE team ran a project to perform the end-to-end technical integration of the service between November 2018 and November 2019. With dozens of participants across most of Loon's engineering disciplines, Loon floated balloon-after-balloon over the Amazon in northern Peru and built an integrated, fully operational testing service.

Notable accomplishments included:

- Turning up Loon's E-Band radios for the first time, in the sky and on the ground, at scale
- Working out timed arrival navigation to deal with adverse wind conditions
- Establishing service measurement criteria
- Adding monitoring and alerting
- Making numerous improvements to Minkowski, Loon's temporospatial, software-defined network (TS-SDN)

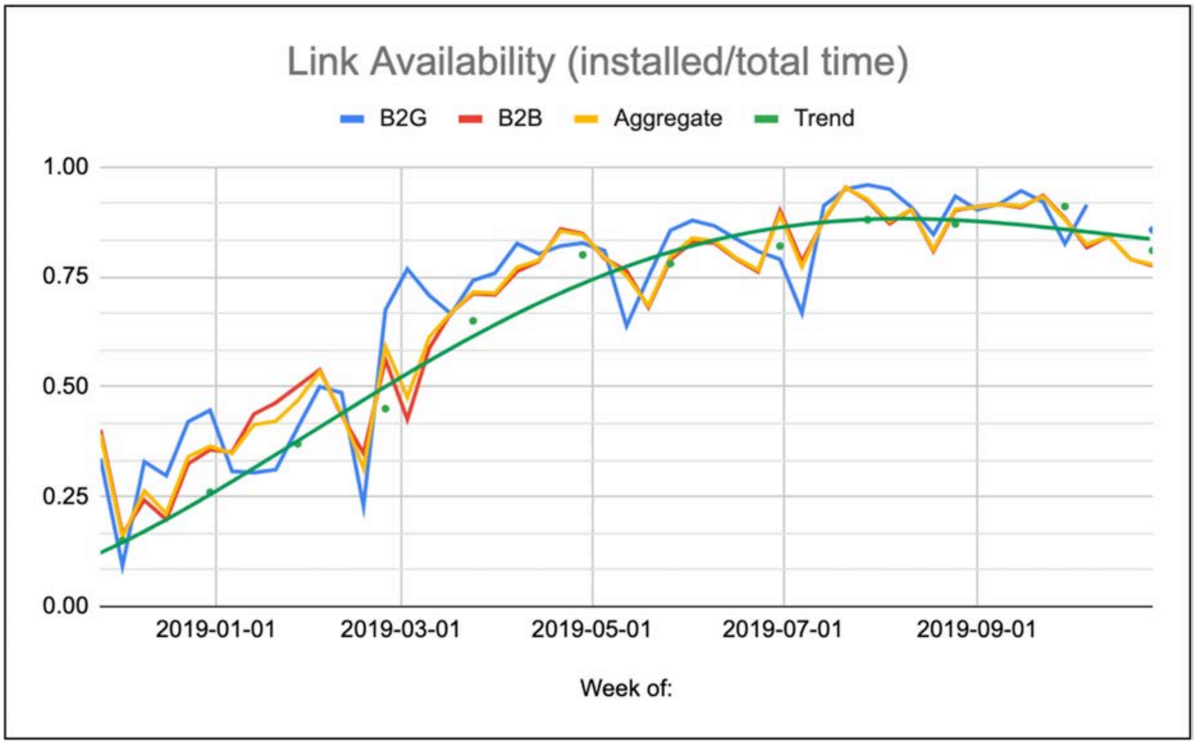


Figure 6-11 ***Debug-a-thon network availability measured as link_installed_time/total_desired_time for balloon-to-ground (B2G) and balloon-to-balloon (B2B) links (also aggregate and trend).***

Key Learnings and Challenges

Loon gained the following essential understanding:

- Production Engineering/SRE team was ideally situated to lead and drive the cross-service integration effort
- Cross-functional approach was critical to breaking down knowledge silos
- Nothing can replace the experience of encountering the real world
- A cross-functional engineering team was born with knowledge outside of their silos (as intended) in addition to an integrated service
- The team should have prioritized Implementing metrics and debuggability functionally earlier on "Start Simple and Iterate Quickly" on page 321

Loon faced the following challenges in the final system integration process:

- Integration process was slower than expected, with progress limited by poor availability of required test balloons due to the small test fleet.
- Prior to in-field testing, up-front, system- and subsystem-level ground-based testing was performed in both super-realtime software-only (full hardware abstraction) and hardware-in-the-loop scenarios. However, a number of issues found during in-field integration showed coverage gaps in that previous testing that could have been addressed with a more methodical approach. In addition to accelerating the final effort, this would also have been more cost effective.
- In-field testing was significantly impeded by the poor debuggability of the system

Applying Engineering Methods to Operations

Loon applied software engineering tools and approaches to managing service operations. We used service availability as the starting point for understanding how the systems were performing. Insufficient on its own, service availability became one facet of an integrated production fleet operations approach that we organized using a forum we called NaaS/Nav (Network-as-a-Service / Navigation). To better understand and address the inevitable failures, we integrated blameless postmortems into the culture of Loon engineering and aviation operations. This critical retrospective analysis tool proved its worth during commercial service deployments consisting of a series of bring-up field tests in Kenya.

Managing Through Service Availability

Loon’s Production Engineering/SRE team organized itself around defining, measuring. and improving Loon’s core service metrucs, most visibly, availability. The team implemented an effective separation of duties with the Flight Engineering team:

- FEs were responsible for safety and flight (e.g., vehicle anomalies, coordination with ATC).
- Production Engineering/SRE was responsible for the mission (i.e., delivering communications services).

In addition, the Fleet Management software team and Production Engineering/SRE teams maintained responsibility for keeping the fleet management systems up and running.

Loon did not have or need a Network Operations Center (NOC) to monitor and manage the mission (e.g., networking and LTE services).[5]

The Production Engineering/SRE team wrote the first-service level monitoring for the system, focusing on implementing symptom-based alerting that signaled when the service was not performing well for users. Loon used both whitebox and blackbox metrics and saw challenges with each. Collecting and using whitebox metrics was confounded by the difficulty of bridging telemetry from intermittently connected nodes.

Black box monitoring was accomplished using a combination of physical LTE service monitoring probes called *Romers*, deployed in the field at specific points of interest, and software-only virtual probes that could accurately approximate service anywhere. Both of these tools are described more in "Service Monitoring Probes" on page 230. In this way, Loon attempted to implement blackbox monitoring that balanced both precision and recall.

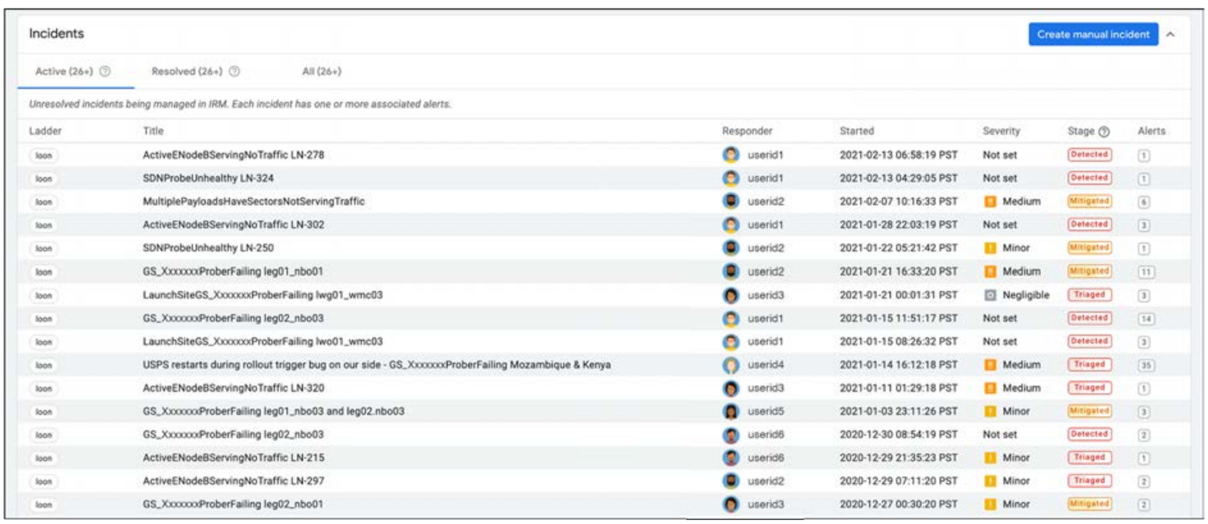


Figure 6-12 Incident management dashboard.

---

5 The service support structure was rounded out by 1) Loon and partner teams focused on in-country hardware, and 2) a Google support team that specialized on providing support to telecommunications partners (e.g., Loon's MNO partner).

Incident response was organized around minimizing the mean time to recovery (MTTR) from the inevitable failures. Loon built a core on-call team that maintained a five-minute response time rotation out of the experienced Production Engineers and a hand-selected set of subject matter experts (SMEs) from Loon's capable development teams. The core on-call team was supported by a set of SME rotations with people who had a greater knowledge of certain complex parts of the system (e.g., Minkowski, PSC, LTE). The response time of these SME rotations was not as stringent as with the prod on-call rotation.

Start Simple and Iterate Quickly

Given the Production Engineering/SRE approach Loon was taking to service management and delivery, a natural place to start was with setting Service Level Objectives (SLOs). What was not at all obvious was what to measure and where to start. Loon knew the goal was to eventually track the top-level SLO for Loon's LTE service offering, but it clearly did not make sense to try to start there before we integrated the system end-to-end. Instead, Loon took a launch and iterate approach, starting with what it could measure immediately and improving/evolving from there.

The Debug-a-Thon work and service in Peru highlighted the fact that the overall service availability had both temporospatial and system reliability components. Loon settled on the following simple formula to start with:

$$\textit{service availability} = \textit{navigation availability} \times \textit{network availability}$$

Although yield-based metrics (i.e., successes/attempts) tend to provide more robust measures, Loon started with time-based metrics.

Loon defined navigation availability as:

$$\textit{navigation availability} = \textit{time in region}/\textit{time desired in region}$$

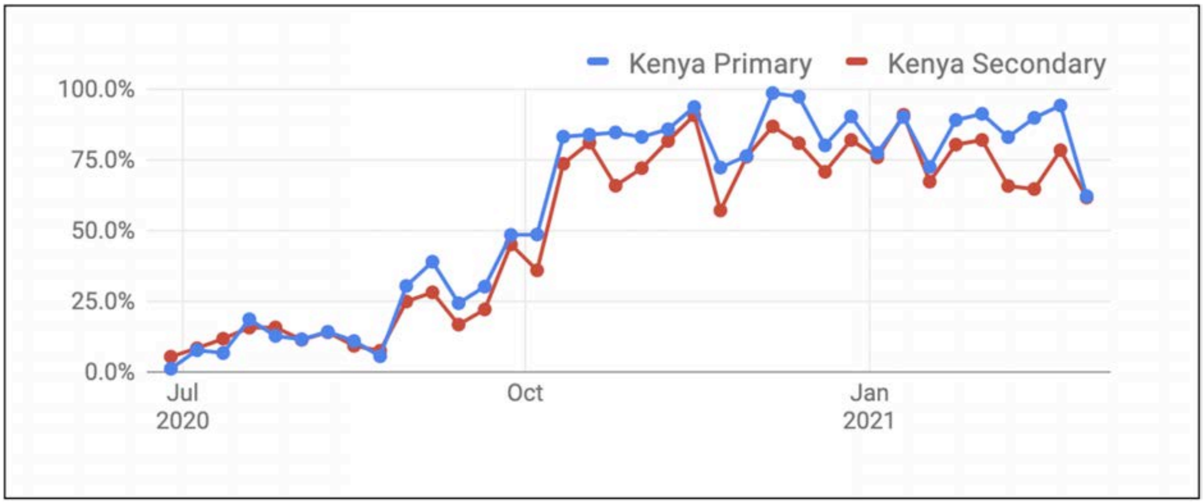


Figure 6-13 ***Navigation availability documenting presence of balloons in the primary and secondary LTE service regions in Kenya.***

Similarly, Loon defined network availability as:

network availability = time serving LTE/time LTE desired

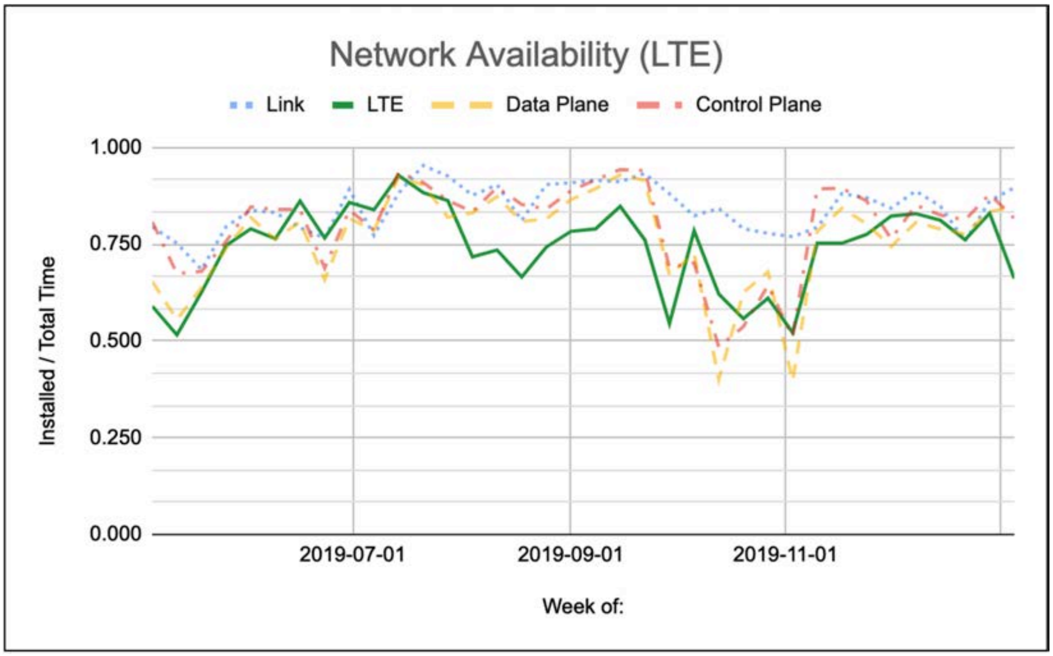


Figure 6-14 ***Network/LTE availability presented alongside independent measures of link layer, control and data plane availability.***

For navigation, Loon realized that time within the service region was only valuable if it could bring up a service network, so Loon refined navigation success to the cases where it was in the service area *and* 1) within range of a ground station, or 2) within range of a balloon that could connect to a ground station. As Loon focused on bringing up the E-Band radios and TS-SDN, Loon defined backhaul availability as the key metric to track. It was defined in the same time-based manner as the others.

network aware navigation availability =
time in region within range of network / time desired

backhaul availability = time with backhaul/time desired

Capability Threshold Identification

In addition to enabling Loon to track progress as it integrated the TS-SDN network and then the LTE system, tracking progress against SLOs also gave the Loon team a principled way to identify the capabilities of the underlying system. That is, while the aspirational goal was to provide a 99.999% available LTE service, it was clear that the generation of wind-navigated platforms and systems Loon was fielding was not going to be capable of hitting that target.

Loon initially set the target backhaul availability at 99.9%, believing it to be one that would be a stretch but could potentially be in reach. As integration work continued, and it layered on the LTE system, Loon saw substantial improvements. Beginning with a backhaul availability of 18% in November 2018, Loon hit 91% in September 2019. LTE (network) availability similarly grew from 6% in November 2018 to a peak of 77% in September 2019. By this time, Loon had harvested the low-hanging fruit of network availability and found that additional improvements were increasingly costly to implement.

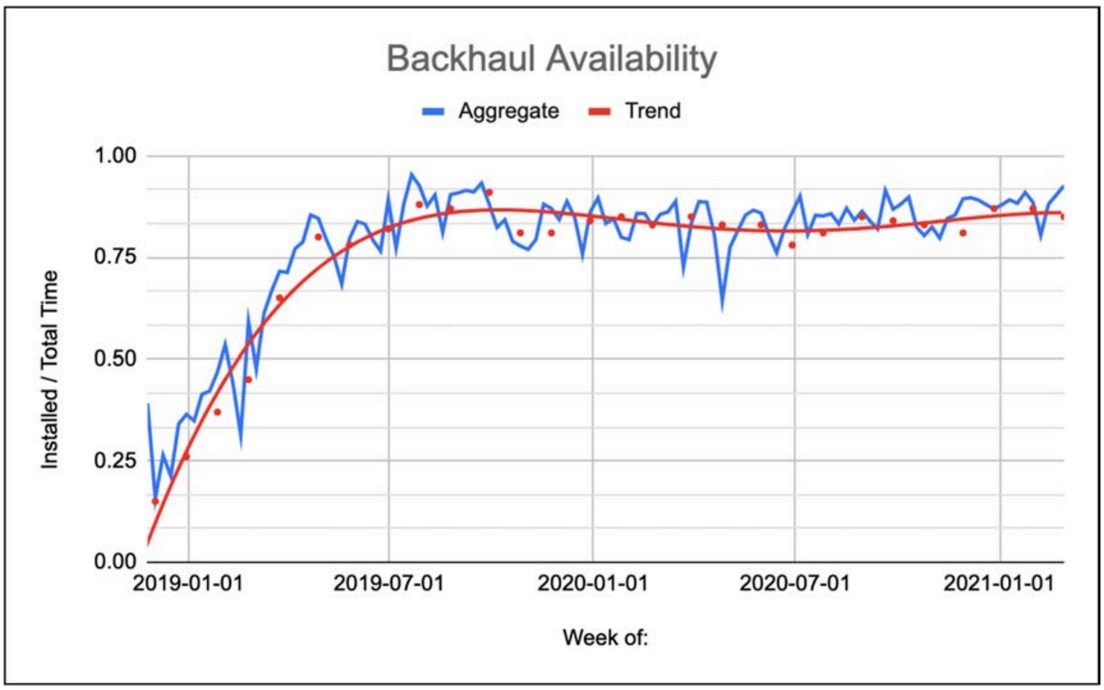


Figure 6-15 ***Backhaul availability settling at a discovered performance threshold.***

As integration work continued, the interaction between navigation and network availability also came into greater focus. What became clear was that Loon's service availability was primarily going to be limited by navigation availability. It was important to ensure that the network came up when the balloons were in the right place, and even more important, to ensure that the balloons were in the right place. This insight enabled Loon to reduce the effort on improving network availability further and refocus efforts on other activities.

Balloon Focused vs. User Focused Metrics

One shortcoming of the availability metrics that Loon never had time to address was that of perspective. The metrics presented above are all rooted in the balloon/platform view of the world. That is, similar to many metrics in data-center-based systems, Loon was using server- side metrics to stand in for user-visible performance.

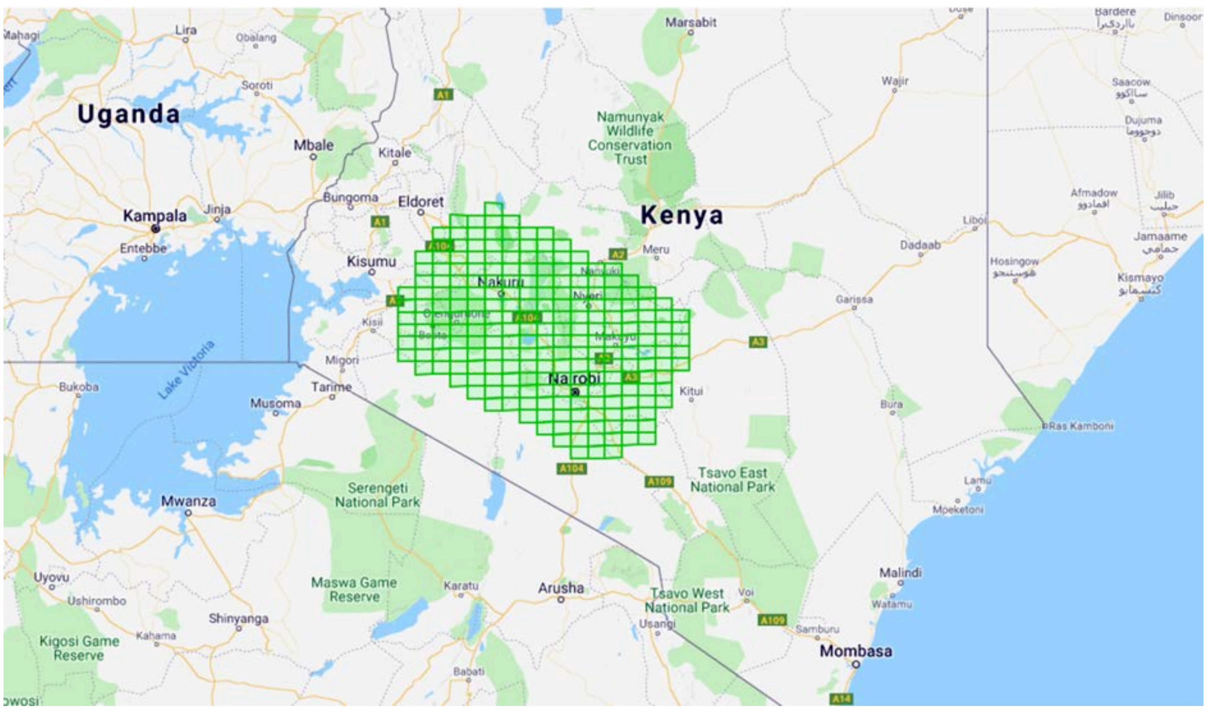


Figure 6-16 ***Provisional service region in Kenya broken down into S2 cells.***

A better approach would have had Loon taking the perspective of a user on the ground. Imagining a user in a fixed location or within a small population center, what would be most important to them would be whether they had consistent LTE service or not.

Loon planned to invert the availability calculations to account for this, computing service accessibility and consistency metrics for each S2 cell[6] in the service area. A natural summary of this data that would align better with user happiness would be a population weighted assessment of availability.

Key Learnings and Challenges

Loon gained the following essential understanding:

- SRE approach worked well for autonomous service deployment
- The combination of physical and virtual probes provided a good blend of accurate measurement (precision) and scale (recall)
- Blended core on-call rotation — Production Engineering/SRE + SMEs — provided a continuous opportunity for learning and cross-training in the rotation
- Start with what you can measure and iterate/evolve from there
- Navigation success meant balloons within reach of the network (ground station directly, or in mesh) rather than just in the service area
- Availability metrics are useful both to track the service integration process and to identify capability thresholds
- Navigation availability dominated the overall service availability in the system

Loon faced the following challenges:

- Debugging intermittently connected systems with no physical access is hard
- Would have also liked to reframe from a geographical and user perspective (i.e., What is the availability experienced by a user in a given location over the period of time?)

Blameless Postmortems

Loon adopted the Google SRE practice of using blameless postmortems[7] as a tool to understand and learn from its (inevitable) failures, which was a natural fit for Loon. The use of postmortems at Loon was particularly fitting given that it represented a return, of sorts, to the original inspiration of NASA's Aviation Safety Reporting System.

6 https://s2geometry.io/
7 https://sre.google/sre-book/postmortem-culture/

The Production Engineering/SRE team introduced tools and structure originally developed at Google. The team produced training, repeated periodically, and formed a postmortem working group whose goal was to ease adoption and promote the use of tools.

Postmortems were common, covering both aviation and software issues. They formed an intuitive meeting point between the reliability culture of the Production Engineering/SRE team and the safety-first culture of Loon's aviation organization.

Key Learnings and Challenges

Loon gained the following essential understanding:

- Postmortem approach and format stretched well across service and aviation failures.
- Postmortem structure proved to be a good learning tool when used in successive rounds of bring-up field tests. Fed into development of repeatable processes.

Loon faced the following challenge:

- Working Group structure (i.e., consisting of volunteers) always left this work underfunded.

Commercial Service Deployment: Kenya Bring-up Field Tests

The Loon team performed a series of field tests of increasing difficulty as a part of bringing up commercial service in Kenya. Coordinated as a distributed team due to COVID-19 restrictions, a dozen Loon engineers would jump online (most at about sunset in the San Francisco Bay Area), to prepare for and manage field tests halfway around the world. Working with the FEs, balloons would be floated in with as much precision as possible to hang out over the targeted test zones.

On the ground, Loon partners would perform testing with handsets and other measurement equipment. Loon started small, working with Loon contractors on the ground in Kenya. From there, Loon began to work directly with Telkom Kenya engineers, who analyzed the service to verify that it was ready for their customers. Loon published a detailed test record for each of these exercises, to understand the shortcomings and to drive improvements to the process. This report also included a retrospective, following the SRE model of blameless postmortems.

The tests culminated with a public event on July 11, 2020. A field test was held for Kenya's Information, Communications and Technology (ICT) Minister Joe Mucheru. The wind, weather, and demo gods were smiling down upon Loon that day, because we were able to pull off this final field test without a hitch, allowing Minister Mucheru to carry out a live video call with Kenya's President, Uhuru Kenyatta. This successful field test unlocked the start of commercial service for Loon.

Figure 6-17 ***President Uhuru Kenyatta announced Loon internet balloons for access to 4G internet.***
The event was at Radat, Baringo County and Presided by ICT CS Joe Mucheru.[8]

Key Learnings and Challenges

Loon gained the following essential understanding:

- Kenyan service deployment success built upon previous cross functional collaboration for service integration in the Debug-a-Thon
- Stepwise validation with increasing partner visibility, and then press visibility was a good approach

Loon faced the following challenge:

- Deployments in Africa meant that this work often ran through the night in the San Francisco Bay Area.

8 https://youtu.be/be8XSE3Aio8

Chapter 7

Interface with the Real World

This chapter covers Loon's challenges, innovations, and key takeaways for deploying into the real world.

We start by considering the deployment process and explore the key challenges that impacted the service deployment process and how these were addressed. We then explore the high-level telecom and aviation regulatory environments which governed how Loon operated and some of the key takeaways.

We then turn to the operation world: Launching, managing airborne operations, and recovering at scale. In particular, we explore the innovative framework implemented to safely supervise a large-scale automated fleet system with limited team size. Loon believes that the principles developed extend well beyond balloons and potentially to many highly automated unmanned aviation operations. Finally, we discuss safety and the innovative framework developed in partnership with regulators that enabled Loon to fly safely at scale while rapidly iterating. We believe that many of the safety principles developed can generalize to many unmanned aviation systems, and in particular other HAPS platforms.

Deployment

Deploying Loon's balloon-based mobile network expansion service into a partner MNO's network involved the coordination of many stakeholders and work streams across Loon, Loon's in-country contractors, the MNO partner, regulators and other government entities. For reasons that we discuss below, Loon's first commercial deployments were more complex and took longer than first anticipated. This section discusses some of the unique challenges that Loon experienced and some key learnings gained along the journey.

Challenges

Complex, Unique, and Unparalleled System

Loon's system did not fit readily into existing regulatory frameworks, standards and technologies. In order to work with established aviation and telecommunications partners to deploy Loon's service, a significant number of custom integrations were required. In addition, the Loon team worked to provide education on technical and operations details about the Loon system to potential and existing partners and stakeholders.

On the telecom side, Loon's system connected with the partner operator's core network, which required a significant amount of hardware installation, software integration and coordination across multiple technical teams. The nature of the Loon system also necessitated closely partnering with Air Navigation Service providers to adapt and enhance traditional air traffic control (ATC) procedures to facilitate the safe operation of the large-scale deployment of stratospheric balloons. Lastly, Loon's system was highly sensitive to a multitude of parameters, including wind conditions, weather conditions, safety limitations, airspace restrictions, service area requirements, ground station positioning[1], launch capacity, etc. A change in any one of these factors could change the expected performance of the system. For example, a bad wind pattern in Puerto Rico could result in a lack of service availability in Kenya one month later, or an airspace restriction could make it harder to provide service in a desired area.

1 Ground station positioning was constrained by multiple factors: ideal theoretical geographical position (from a system resilience and throughput perspective), access to power and high speed fiber, unobstructed view of the sky 360°, physical security. Once all these factors combined, finding suitable locations could be challenging in some areas of the world.

Transition from R&D-product readiness

While rapid progress was made over the years, the transition from a proof of concept or demonstration phase to a full-fledged commercial operation required more time and resources than originally anticipated. These challenges fell into a number of buckets:

- Technical readiness: Loon had made tremendous progress in the longevity of its balloons, but achieving consistent, high reliability across all subsystems on longer airborne durations was difficult. Systems with moving parts or advanced wireless radios were particularly challenging. Loon's encounters with meteorological phenomena[2] and stratospheric weather events often required the development of new technology, procedures or restrictions, which added time and difficulty to commercial deployments.
- Operational readiness: Flying at scale and deploying commercially also required new operational considerations. For example, overflight agreements and spectrum licenses obtained for R&D purposes would need to be renewed or updated for commercial service. Existing aviation procedures and technologies required innovation and adaptation to support the increased scale of Loon's operations. Landing, recovery, export, and recycling of large numbers of vehicles needed a more robust infrastructure and specific authorizations, and customer-oriented support required an infrastructure that needed to be developed.
- Organizational readiness: Shifting from rapid innovation to commercial service meant building new levels of cross-functional coordination, creating new functions and expertise, and stabilizing and ramping production on optimized product designs.

Coordination of multiple authorizations with regulators and partners.

Loon's technology operated at the intersection of two of the most heavily regulated industries in the world — telecommunications and aviation — with regulatory oversight at the national, regional, and global level. As a result, successful and timely deployments depended on obtaining and coordinating regulatory approvals for many aspects of our operations, including launch, landing, and overflights, import and export of equipment, and spectrum licenses. This coordination was different from country to country and was an area of significant focus for the team, sometimes resulting in unpredictable timelines.

---

2 Most of these phenomena and their consequences could not have been anticipated without flying at scale and starting to deploy the service.

Regulatory

Telecom Regulatory - Key Challenges

The Choice of E-Band for Backhaul Spectrum

Based on its narrow beam width (a fraction of a degree), wide bandwidth (multiple GHz), and relative underuse in rural environments, Loon chose E-Band as its backhaul spectrum for ground-to-balloon and balloon-to-balloon backhaul networking. However, the fact that there are few established E-Band regulations for Loon-like applications meant that deploying an E-Band solution was not straightforward.

Lack of National Regulations to Support Loon.

Because no HAPS service had deployed commercially before Loon, national regulators had not created licensing frameworks for HAPS services. As a result, Loon had to work closely with regulators in its service territories to develop regulatory frameworks to support Loon's service. While many regulators were eager to develop these frameworks, the process and timeline for implementing them was often uncertain.

Equipment Authorization and Import/Export Delays.

The equipment import and use authorization process can be time-consuming, particularly with new technologies and those requiring extensive lab testing. Even after equipment is authorized, delays in import and export can set back deployment.

Changes to Loon's Product and Deployment Strategy Required Continual Education.

Over its life, Loon rapidly iterated its technology and concept of operations. While these developments resulted from significant technical advances, many of these required re-educating regulators about the technical details of the Loon system.

Telecom Regulatory - Key Learnings

- To scale a business that is designed to be international, a comprehensive strategy that promotes regulatory harmonization at the regional level is essential.
- Regulatory timelines are often unpredictable. The process can take months or years, or may never conclude. Overcoming these barriers requires a combination of on-the-ground presence, coalition advocacy, and supportive regulators and governments.
- While International Telecommunications Union (ITU) cycles are long and uncertain, the failure to participate actively in ITU proceedings can impose significant regulatory burdens on deployment, particularly in markets that largely defer to ITU radio regulations for internal regulatory frameworks.
- Inflexible service and technology definitions, particularly in treaty-level regulations such as the ITU Radio Regulations, can be a significant innovation inhibitor. Businesses and regulators should partner in promoting flexible service and technology definitions so that the best technology can flourish, not the technology that best meets existing regulatory definitions.
- There remains a significant opportunity for regulators to promote digitization and automation of spectrum management, particularly in millimeter wave bands. This will be particularly important for the coming wave of non-terrestrial networks, which will need modern spectrum management systems to promote coexistence.

Aviation Regulatory - Key Challenges

Lack of relevant regulations

There are no existing aviation regulations to adequately govern a system such as the one Loon was proposing to use. To compensate for the absence of globally accepted safety standards and means of compliance that would ensure the safety of such operations (and facilitate regulators' assessment of the operation), Loon focused on the spirit of traditional aircraft operations and implemented a Safety Management System (SMS) that provided a strong baseline of safe operations and included a framework for continuous monitoring and improvement.

Country-by-country approvals

Because of this lack of regulatory framework for a large-scale autonomous balloon operation, each country where Loon needed to launch, overfly and/or land balloons required specific permission and a documented Letter of Agreement outlining the scope of permissible operation.

Aviation Regulatory - Key Learnings

- The lack of an established regulatory framework presented both opportunities and challenges. On the one hand, it enabled Loon to demonstrate the safety of a technology and operational scale that had never been accomplished. It also allowed Loon to demonstrate novel concepts for human to machine interactions such as Attended Autonomous Fleet Management (see "Airborne operations at scale" on page 339) which had never been explored in unmanned aviation. On the other hand, it made engagement and approvals disproportionately dependent on individual change agents within the government.
- The process of establishing safe operational procedures in the absence of well established approval paths made establishing accurate timelines particularly challenging. Slow, steady engagement with full transparency allowed eventual approvals, but the time it took to do that varied widely by country.
- The development of a unique Concept of Operators for the Stratosphere demonstrated that the industry was willing and able to work as one, promoting best practices for the sector and greater aviation.
- Working with ICAO and using its regional venues and aviation forums was crucial to promoting Loon's operations.
- In the absence of international regulations, a solid and self-driven Safety Management System was critical to demonstrate Loon's commitment to safe operations.
- Partnership with advanced aviation authorities such as the FAA, EASA, Transport Canada, and CASA, helped establish legitimacy, develop new frameworks, and promote global interoperability.
- Developing early relationships with the CAAs, which became known as early adopters and advocates of Loon's safe operations was very helpful. It is invaluable to have highly-respected organizations and individuals championing your work on your behalf. Engage those partners early and often to tell your story.

Launch

Loon operated two launch sites: one for R&D and production (commercial service) flights at the Winnemucca Municipal Airport Industrial Park, in Winnemucca, Nevada, and a production-flight-only facility at the former Roosevelt Roads Naval Air Base near Ceiba, Puerto Rico.

Key Challenges

Many moving pieces

Loon was continually iterating and improving many elements of the balloon, bus and payload and these changes had ongoing changes to their pre-flight and in-flight test processes. This complexity forced the establishment of a difficult and time-consuming process to synchronize and ensure that all elements required for a successful mission were present and working[3]. Incremental changes to the communications payload or service-oriented software did not always get adequate regression testing for pre-flight validation but fortunately those systems were increasingly able to be updated in-flight. Similarly, each R&D launch featured slightly or significantly different flight system configurations, making it tough to assess the particular configurations that led to a successful test.

3 Flights were always flown safely due to redundancies and thorough safety checks. Some failures (e.g. gimbal) could however incapacitate a flight to deliver connectivity. Most of the time (when safe to do so), the flight would remain airborne and be repurposed to another objective (e.g. improve steering algorithms), however those failures would prove to be costly.

Key Learnings

Launch site location matters

The Launch site selection had to balance several factors to enable launchability throughout the year at affordable operational cost. Puerto Rico's location near the equator allowed Loon greater access to variable winds that could take balloons to service areas in Africa, Latin America, Australia, and elsewhere. As a US territory, Puerto Rico also presented business and operational advantages; in particular shipping, import, and aviation regulations (under FAA rules) were aligned with Loon's California- and Nevada-based facilities and operations. The trade-off was that the humidity and high temperatures of the tropical climate affected the balloon materials, which (if not extremely well stored and handled) could reduce flight durations. Winnemucca's value stemmed from its relative proximity to Loon's R&D team in Mountain View, low population density in surrounding areas (useful for short duration test flights and easier recoveries that enabled post-flight analysis), and supportive city and county administration. Winnemucca's challenge was that wind patterns in December and January of each year were unfavorable and risked carrying balloons into regions where Loon did not intend to fly, reducing the number of viable launch days.

Automation promotes standardization, scalability, and safety

To decrease this workload and increase the speed, safety, weather window, and repeatability of launches, Loon developed the Portable Launch Rig (PLR). The PLR was a boatyard gantry crane modified to add wind walls along three sides, hoist the balloon, fill it with lift gas, and then release it with the press of a button controlled by an operator 20 to 30 yards away. This shrank the number of launch personnel needed by a factor of three to four (down to about five people), and balloons could be launched every 45 to 60 minutes in good conditions. Surface weather remained a limiting factor, but to a much less significant degree: the PLR's wind shielding allowed for launches in a greater range of wind speeds.

Figure 7-1 ***Launching with the Portable Launch Rig (PLR)***
The PLR significantly improved launch scalability, reduced the need for launch personnel, and was a key element enabling long flight durations due to a more delicate and consistent handling of the balloons.

Launch automation further reduced the risk of contaminants such as the oil from human hands, and improved the consistency and delicacy of handling, all of which contributed to lengthening the lifespan of balloons.

Enforce design change processes

From a commercial deployment perspective, Loon would have been better served coordinating design roll-outs across all subsystems, and more strongly enforcing design change processes, although this would have slowed the pace of development.

Mitigate launch risks with mock launches

To mitigate the risk posed by newly introduced R&D flight vehicles, Loon conducted mock launches (in the large hangars at Moffett Field) and/or tethered[4] launches. These helped rethink wind limits, launcher clearance, or gas fill issues and provided training opportunities for the launch team.

Clearly define roles, responsibilities, and decision makers

The prep, launch, and flight of balloons require many different roles performing a wide variety of tasks. Likewise, process design and troubleshooting pre-flight anomalies involved technical and operational experts from across all parts of Loon. Ensuring the efficiency of this collaboration required two things: 1) shared understanding of every team member's role and responsibilities, and 2) a crystal clear escalation path and decision-making framework.

Promote a strong safety culture

As is expected of any industrial site operating heavy machinery, the development and indoctrination of a strong safety culture are critical.

4 Full launches of systems that remained tethered to the ground after launch.

Airborne operations at scale

Key Challenges

Supervise worldwide operations 24/7

The continuous operation of a large fleet of vehicles that overflies air traffic routes and ground populations requires[5] active 24/7 supervision. This is first and foremost to guarantee the safety of operations and to ensure the system provides optimal service and functions as expected.

Figure 7-2 ***Snapshot of the Loon Airborne Fleet.***
Loon's airborne fleet reached over 90 simultaneously airborne vehicles. At any time, the fleet could be distributed over more than ten countries or Flight Information Regions (FIRs). A typical balloon flight would overfly fifteen unique countries, sixteen unique FIRs and cross country borders over 200 times.

5 This is not a regulatory requirement. Most balloon regulations (in particular ICAO Annex 2 Appendix 5) do not specify any supervision requirement.

Integration within the existing Air Traffic Management system

Loon had to determine how a large-scale, mostly automated, and highly dynamic system with probabilistic behaviors, could safely and efficiently integrate within the existing airspace management and air traffic control frameworks. Those frameworks rely on radio voice communications between pilot and controller, and are designed primarily for short duration flights that navigate from pre-known starting and ending points in a deterministic, fully controllable manner, with piecewise linear paths.

Key Learnings

Use an exception-centric approach, not a vehicle-centric one.

Unlike traditional aviation, which uses a vehicle-centric framework (i.e., a pilot and crew are assigned to a specific aircraft), Loon's airborne operations relied on an exception-centric framework. In this framework, no human or team was assigned to supervise a specific vehicle, group of vehicles or region of operation. Instead, humans were dynamically assigned to handle exceptions as they arose. An exception was defined as a deviation from normal operating parameters, and could include things like low helium or power levels, a flight path toward a restricted airspace, etc.

Automated exception detection rules were at the core of Loon's exception-centric approach. These were implemented in a software system that would raise the exceptions, assign a priority and severity, and alert the Flight Engineers on duty. Any flight engineer could accept the exception and become the owner of its resolution. Continuous monitoring, maintenance and improvement of this exception detection service was fundamental to the safety of operations.

This approach to airborne fleet management is similar to approaches used in satellite operations, power grid management, nuclear power plant supervision, and data center management.

Implement design principles that promote safety

The systems need to be redundant, modular, decentralized, and degrade gracefully, since a system-wide outage would pose a greater risk than an outage of any one individual flight vehicle (see more in "Graceful Degradation" on page 298). Continuous and holistic system supervision and maintenance is paramount. This includes the fleet, onboard automation, and ground automation. When an automated system fails, the priority is to fix it immediately (or to apply a mitigation to limit the failure's impact in the short term), rather than attempting to manually perform the actions that would otherwise be automated.

Automation monitors, humans supervise

In aviation, humans have traditionally been assigned to monitoring tasks (for example, a pilot scans the horizon to monitor for traffic that would not have been identified by ATC or collision avoidance systems).

Loon's philosophy was different. Loon believed that monitoring tasks should be the responsibility of automation[6] which would monitor the incoming data streams and raise exceptions; thus freeing humans from the need to analyze large amounts of data[7] and remain constantly alert, searching for possible exceptions.

Establish a decision escalation chain of command headed by the Operations Director

In aviation a pilot has full authority over a specific aircraft, and his or her scope is limited to that aircraft. In Loon's system-wide supervision framework, operational decisions could affect an entire region, group of vehicles, or even the entire fleet. Judgment calls would necessarily need to balance a number of factors. In lieu of a Pilot in Command function, a well-established and scalable escalation chain was essential to ensure that the most critical decisions were performed at the appropriate level of responsibility. This is like other large operations management such as power grid management, military operations, and space missions.

6 Humans have short attention spans, and are unreliable identifiers of off-nominals (especially with large amounts of information).

7 Loon's system generated too much information for humans to process (thousands of data points were generated by each vehicle every minute or few seconds, terabytes of weather nowcast and forecast data was regularly ingested along with worldwide airspace data.)

Staffing should be a function of workload, not fleet size

Teams were sized and staffed relative to the workload they performed, rather than relative to the fleet's size. In unexpected surges of workload, off-duty flight engineers could be paged to help to absorb the increased workload. This was made possible by Loon's secure and robust cloud-based infrastructure and web-based user interfaces that permitted any authorized personnel to assist from anywhere in the world with a reliable internet connection, adequate equipment, and a good working environment.

Act on automation

Flight Engineers were encouraged to act on automation (rather than the vehicle) wherever possible, which, in turn, would act on the vehicle (sending commands, monitoring telementry, etc.). Flight Engineers could act on any level of automation but were encouraged to act on the highest level possible, leaving automation to handle sub-level tasks. For example, by modifying the dispatcher configuration instead of manually assigning an objective (steering controller) to the vehicle, or sending navigation commands to the vehicle.[8]

A key mission of the flight engineers could be thought of as automating themselves out of a job. Constantly improving operational processes, tools and automation was essential to improving scalability. Tracking when, why, and how humans intervened or overrode automation, was central to continuously improving the system and enabling more extensive operations.

---

8 Flight Engineers had many methods for intervening in automation's decisions. There are roughly three levels of controlling automation from a navigation perspective, which are described in detail in "Navigating on the Wind" on page 255:

- Navigation commands were the individual commands sent to the vehicle's avionics. For example, a command to move to and maintain a specific altitude.
- Steering Controllers (ground-based algorithms) send such altitude commands to control navigation, based on what the controller is intended to achieve–such as heading, speed, navigation to a point ASAP, arrival at a location at specific time, station keeping, avoid a storm or an airspace, etc.
- Dispatchers (ground-based algorithm) orchestrated the fleet mission assignment to meet overall service objectives. Dispatchers assigned steering controllers (i.e. specific tasks / missions) to individual vehicles. Dispatchers would also assign vehicles to recovery missions when necessary or assign safety objectives (e.g. prioritize avoiding the storm)

Recovery

Loon was committed to ensuring the safety of everyone and everything on the ground and recovering all balloons that could be recovered without endangering anyone. To this end, Loon worked to identify suitable landing zones in strategic locations around the world, as well as reliable local partners able to assist with the recovery and end-of-life processing of landed systems. The Loon Recovery team worked closely with Flight Engineers and a global network of recovery partners to direct the safe landing of every domestic and international flight.

Key Challenges

Mitigating landing zone risk

Landing zones required time, effort, and money to resource properly, and landings themselves presented safety and reputational risks that increased as Loon's landing zones diversified.

Geographic diversity is preferable but not always possible

The intersection of ideal landing zones, strong recovery partners, and complete regulatory approvals were not always easy to achieve and could make identifying geographically distributed landing zones challenging.

Maintaining information networks requires planning, time, and continuous effort

Gathering reliable information about ground conditions on short notice could be difficult. It was essential to build diverse information networks that could be tapped into rapidly, when needed.

Key Learnings

Balance of landing zones

Determining the appropriate number of landing zones needed to support operations was a delicate balance. Too many landing zones could present logistical, operational, and scalability challenges. Conversely, the lack of sufficient, reliable landing zones could increase the risk of not having an adequate landing zone when needed, preventing Loon from scaling its commercial activities.

Fewer, better landing zones – but with contingency plans

Loon preferred to concentrate investments in a few locations carefully selected to meet preferred criteria, rather than pursuing a broad network of smaller sites. However, the reality of working with balloons – even highly sophisticated, self-navigating balloons – required Loon to be prepared for instances when unexpected engineering or environmental issues required us to land sooner than originally planned.

The ideal criteria for landing zones remain the same anywhere in the world

A good landing zone offers predictability. In order to land a high volume of balloons, it was important that Loon was able to count on the area being reliably safe and accessible without needing to reconfirm conditions before each landing. In the future, as Loon increased the precision of targeting and established routine service patterns, we had hoped to move toward an airport model where land was leased and staffed by full-time crews.

Figure 7-3 ***Landing zone in Southern Peru***

Learn from your misses

Loon kept a running analysis of every landing that missed the target or had some potential improvement (e.g., system off-nominal performance, wind causing the system to drag after landing, or inaccurate information about ground conditions). Loon used these learnings to alter landing zones when possible, change the landing procedures, decrease the acceptable operational risk, or give feedback to engineering. When possible, recovery vendors also shipped components back to Loon's California headquarters for detailed post-flight analysis.

Secure permissions from the stratosphere to the ground

Loon sought the support and permission of landowners and surrounding communities to use the property for landing.

Build your recovery network

Recovery vendors were responsible for recovering landed systems, short-term storage of balloon components, conducting analysis under the direction of a Loon engineer or Operations team member, proper disposal of all components, and potentially liaising with local authorities, if needed. There was no set profile for a recovery vendor. Loon worked with a reliable but diverse array of vendors, including safari/adventure tourism companies, logistics providers specializing in multi-day jungle treks (for tourism or research purposes), bird watching guides, and nature conservancy rangers.

Figure 7-4 ***Recovery of a Balloon in Kenya***

Figure 7-5 Balloon recovery in Peru

Prepare emergency response plans for a variety of scenarios

It's critical to be prepared for worst-case scenarios. A good emergency response plan anticipated the unlikely scenario in which a landing caused damage or a negative public impact, identified a clear cross-functional escalation path for each scenario, and went through periodic drills for these scenarios.

Aviation Safety

Safety was Loon's foremost priority. Loon went far beyond what was required by regulators; it implemented a Safety Management System (following ICAO's guidelines), developed a strong safety culture, and developed new safety concepts and technologies.

Challenges

Extended operations over population and air traffic.

Loon operations were inherently less risky than passenger transport because no passengers were on board. But as with any aviation application, flying many vehicles, airborne 24/7, over ground population and busy air traffic routes, carries an inherent risk that needs to be managed and mitigated.

Safety metrics and standards not designed for unmanned aviation.

A critical element of any aviation safety case is to first establish a Target Level of Safety (TLS). The TLS defines the maximum acceptable likelihood of a catastrophic event or fatality that must not be exceeded. For manned aviation, regulators have traditionally measured risk and defined TLS per flight hour. For example, the risk of aircraft running along parallel routes is well suited to a risk metric of fatal accidents per flight hour. A 1.5×10^{-8} per flight hour value has been used based on societal acceptance of risk. Those standards are vehicle-centric; they are well suited for manned aviation[9], but do not address the needs of unmanned aviation. When no people are on board, safety considerations are exclusively for exposed third parties (people on the ground, or air traffic underneath). Using vehicle-centric metrics could lead to underestimating or overestimating the risk and could mislead design choices.

Changing operations conditions

The very nature of wind navigation and Loon's dynamic system meant that the path taken by a balloon to get to its destination would change on a regular basis. This meant that overflown population densities, and aircraft routes would change, and that Loon could not measure the safety of its operations in a fixed and deterministic manner (e.g., like the safety assessment of a new air route).

9 Their main objective is to ensure the safety of onboard passengers.

Hybrid and novel system

Systems like Loon's, and more generally High Altitude Platforms (HAPS) are novel systems that do not neatly fit into an existing aircraft/regulation category. They are leveraging new technologies often with no existing design standards, evolve rapidly, and operate in a not well-understood environment. As a result, most certification requirements and associated means of compliance are not adequate (for example– turbulence requirements would assume levels of turbulence experienced at 30 kft. and their impact on transported passengers).

Rapid but safe innovation

The technical challenges tackled by Loon were highly complex and advances required a significant amount of flight data in real operating conditions. Getting airborne safely and early, and iterating rapidly were essential elements to the development of the technology.

Key Learnings

Implement a Safety Management System (SMS)

Developing and implementing an SMS was not required by regulations for Loon Operations, but it was a pivotal contributor to the success of conducting flight operations with more than 1.9 million flight hours. Loon adapted the ICAO 9859 guidance to fit the needs of a stratospheric balloon commercial services company scaling operations. One of the most tangible benefits to developing an SMS was that it was immediately recognizable and trusted by regulatory bodies worldwide.

Create a healthy aviation vs. tech/engineering tension

Taking the most useful aspects of traditional aviation practices while minimizing unnecessary bureaucracy is paramount to going fast, but safely.

Have an Incident Response Plan, and perform drills

Appropriate and timely crisis response to failures or incidents is essential for a globally deployed unmanned fleet of air vehicles. A Loon Aviation Emergency Response Plan was developed to deal with such emergencies. Periodic training and exercise drills were conducted to ensure all appropriate team members were properly prepared.

Chapter 8

Loon's Trajectory and Future Impact

Throughout this Loon Library, we've talked about Loon's mission and history and the many areas where Loon advanced the state-of-the-art or changed people's sense of what is possible. Those earlier sections talked almost entirely about what Loon actually built, flew, and operated. In this next section, we will discuss work that Loon did not complete. Much of this work was in the very early stages of ideation. Some of it reached a testing phase, while some was still more theoretical in nature. As a project with such a heavy research and development background, Loon was always exploring new ways to do things. What is outlined below is a summary of these R&D projects within Loon, as they existed at the time of the company wind down.

This R&D work falls into four principal areas:

- *Transitioning from a free-floating balloon to a unique stratospheric airship architecture with propulsion and ability to carry a higher-capacity payload*
- *Exploring a switch from helium to hydrogen, a more sustainable lift gas*
- *Investigating methods to dynamically allocate spectrum*
- *Working with industry and regulators on a collaborative traffic management approach for stratospheric aviation*

While many of these projects showed potential and held exciting possibilities for further testing and refinement — as is typical of early-stage R&D work — more work is required to prove out usefulness and feasibility. By including this work in the Loon Library, we hope that others will take these ideas and move them forward.

Transitioning to Stratospheric Airships

Justification for Stratospheric Airships

As this document has demonstrated, economically delivering communications services from the stratosphere is a difficult proposition. Although the potential coverage footprint of a single vehicle initially makes the economics look good, the costs related to providing that service are higher than many expect, and the revenue is lower, without a capacity-optimized system design.

As described in the previous sections, despite moving stratospheric technology forward on many fronts. Loon was focused on improving its service in several ways:

- Each balloon is only over a service area and turned on for a small fraction of its lifetime, so the average vehicle utilization was low and could be significantly improved.
- Revenue is significantly constrained by payload capabilities, even when a balloon is serving users

Loon's work over the past few years shows that pivoting away from the superpressure pumpkin balloon to a propelled airship appears to make stratospheric communication services profitable and attractive. The availability and vehicle utilization can be improved by switching to a much larger, more streamlined (aerodynamic) balloon, with propulsion capable of moving the vehicle through the air at a moderate speed while retaining Loon's advances in altitude/wind-based navigation. Such a vehicle could also carry a larger payload, capable of delivering much more capacity and, therefore, earning more revenue.

Internal estimates suggest that such a vehicle would cost 5x to 10x as much as Loon's latest pumpkin balloons but would shrink the fleet size by an even larger factor and, because of increased utilization and increased capacity when utilized, it may earn about 10x to 50x as much revenue during its lifetime.

Prior Work

Flying propelled airships in the stratosphere is not a new thought. Many companies worldwide have pursued and continue to pursue such efforts for communications, disaster relief, earth observation, and other services. Unfortunately, the approaches taken so far have not worked out for several reasons:

- Increasing the size of an airship to go a little faster or carry a bigger payload requires stronger (and heavier) materials and creates more drag, requiring more solar panels and batteries to reach the same speed, which requires more mass, and an even larger airship. This is similar to the decreasing benefit curve, described in "Loon Technical Overview" on page 25, but is even worse for airships because increased size also increases drag (not a factor for un-propelled balloons).
- To have a high availability (>99%), some very high speeds are required because, on rare but not rare enough occasions, the winds are moving extremely fast and that typically happens for hours to days at a time. Even if the vehicle only sees wind speeds of 10 m/s or less during 95% of its lifetime, it may experience 20 m/s, 30 m/s, and higher wind speeds some of the time. These occasional peak speeds require significant over-build of the vehicle to sustain those high speeds for the hours needed to attain the desired high availability.
- For an airship to move twice as fast, around eight times as much power and energy are required. This is actually a v^3 relationship, so twice the speed (2v) translates to $(2v)^3 = 8v$. Moving from 10 m/s speed to 30 m/s requires $3^3 = 27$x as much power and energy (approximately; there are many variations here).
- Due to the extreme difference in air pressure between ground level and the stratosphere (about 20:1 ratio), stratospheric airships are almost nothing like their low-altitude siblings, which only support about 2:1 pressure ratios. Many organizations that start off thinking that it's just bigger soon begin to understand the significant challenges in landing such a large vehicle and the sizable additional cost and complexity that this landing creates.

For these reasons, many organizations have considered, pursued, and eventually rejected the prospect of high-availability stratospheric airships and have instead focused on fixed-wing aircraft or given up on the stratosphere entirely.

Improving the Economics with Loon's Advances

Such an airship can be less expensive and attain even higher availability and vehicle utilization when combined with Loon's advances in altitude-based navigation systems. That high availability does not require the high speeds that drive up the size and cost of the vehicle if the design can take advantage of winds going in different directions at different altitudes. Even a 30 m/s day could be 100% available using a 10 m/s vehicle if that vehicle alternates between wind layers moving in different directions.

Even if the accessible winds are all going in a similar direction, having the ability to move up and down allows the vehicle to find and use the wind layer with the lowest speed wind. Where a fixed-altitude vehicle might be constrained to operate at a layer that has 30 m/s winds, a flexible altitude range would often allow it to find a layer going 25 or 20 m/s.

However, there will still be occasions where the winds are fast and aligned, when no algorithm can keep the vehicle from drifting away from the service target. In these cases, Loon's advanced fleet-wide orchestration (Dispatcher) algorithms, combined with Loon's long-range navigation systems (Cartographer), would allow a small group of vehicles to form a loop through the region. Each would pass slowly by the service area, serve, and then continue around the loop while the following vehicle flies through and serves. Even a modest amount of lateral speed (3 m/s) significantly reduces the fly-around time for such loops and 10 m/s even more so. The number of spare vehicles necessary is therefore reduced by a similar factor; shrinking the 30-day loop to six days could reduce the fleet size by 5x. In addition, because of the lateral speed, they each spend more time in the service area as they fly over, increasing the vehicle utilization.

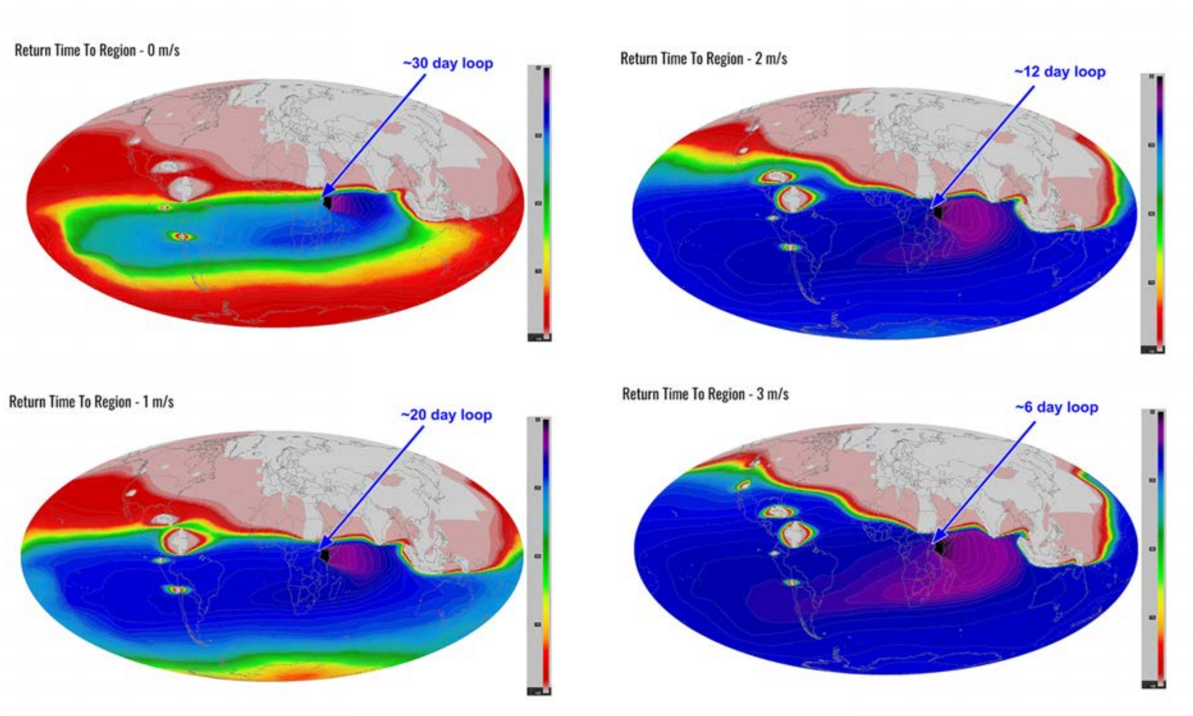


Figure 8-1 ***Reduced return time to region.***
As propulsion goes from 0 m/s to 3 m/s, return time to Kenya goes from an average of 30 days to an average of six days.

One additional advantage of such a vehicle is that, with the increased availability, the service becomes much more attractive and therefore may encounter a beneficial super-regional effect. This kicks in when a much broader region is being served (e.g., encompassing Kenya, Uganda, and Tanzania rather than just Kenya). The vehicles can then earn revenue for nearly all their serviceable life, even during the periods where they're following these worst-case multi-day loops around the area.

Pays for Itself

The downside to this increased altitude range is that it doesn't come for free. An airship designed to operate only at a specific altitude (e.g., 20 km) is smaller, lighter, and therefore cheaper than one capable of controllably and safely moving up and down from 20 km to 22 km. Moving that ceiling altitude requires significantly more balloon volume to provide the same lift. Similarly, extending a fixed-altitude airship's operating floor, say, from 20 km to 18 km, while leaving the ceiling at 20 km, requires the ability to push into the balloon sufficient ballast air to descend to that level, which requires the structure of the envelope to be stronger to handle the higher superpressure.

This requires a higher mass envelope, reducing the payload carrying capacity and/or requiring a larger envelope. *There is no free lunch.*

In addition, the use of that additional altitude range requires sufficient ACS (turbopump) capability to move air into and out of the balloon fast enough to make those altitude changes worthwhile. Loon was planning to use a significantly larger ACS in the airship than had been used in the pumpkin balloons, capable of an airflow rate two to five times as high as the Thor model. In addition, it planned to use at least four and, for some configurations, potentially 12 or more of these units, both to attain the required airflow rate and for redundancy/reliability. (The final determination of total airflow rate and the sizing of individual ACS units was not yet complete.)

Key Learning and Innovation

Loon gained the following essential understanding:

- A larger, aerodynamic airship with moderate propulsion could have much better economics than the pumpkin balloon architecture by:
    - » Carrying a much more capable service payload and so earning much more revenue per service hour
    - » Staying in the service area for the majority of the vehicle's lifetime, thereby serving many more hours
    - » Solving the availability challenges that made Loon an unattractive service for many markets

Systems Engineering of Airship

How do we determine if extending an airship's operating ceiling or floor is sufficient to reach the targeted service levels or service costs? If so, how far should it be extended, and what should the propulsion speed be? These many factors of altitude range, balloon size, cost, propulsion speed, payload mass, and power are highly interdependent and intrinsically tied to the nature of the winds, the required availability, and the type of service. As a result, it is impossible to do a back-of-the-envelope calculation to prove that a particular service level and cost-of-service is possible with a particular candidate design. To go to the next step and determine the optimal design and sizing for a new vehicle (including the altitude floor, ceiling, envelope size, propulsion speed, and payload capacity) requires a full suite of tools even more extensive than those Loon developed for its pumpkin balloons.

These new or enhanced tools consisted of the following:

- Creation of a more extensive version of the Loon dynamic vehicle model (VSFEI) to incorporate all the relevant differences between the airship and the pumpkin balloon architecture, including the propulsion and aerodynamics, controls and stability, and solar panel placement.
- Major enhancements to the static vehicle model within the Vehicle Sizer/Optimizer tool, again, to incorporate the airship's many differences.
- Additionally, the Sizer needed to quickly grade the steering performance of the thousands of new design candidates that it would test during an optimization run, without running full fleet simulations. That method was known as the *Extremely Simple Steering* model.
- The full simulator, VSFEI, also needed to incorporate new navigation algorithms that take advantage of propulsion and altitude-changing capabilities.

The tradeoffs between these design variables and the potential profit of the vehicle must be analyzed on a per-region basis due to the extreme variability in the winds. This requires simulation of each candidate vehicle design across each region's historical wind data. The overprovisioning of the fleet and the fleet utilization (percent of time vehicles are earning revenue, on average, for all the vehicles in the fleet) can only be determined through such an analysis.

Hammerhead

Building these tools and performing this analysis is the work the Loon System Engineering and Product teams have been doing over the past two years, leading to our planned future roadmap of airships, starting with a 60 m-long vehicle, code-named Hammerhead, with characteristics shown in Table 8-1. Such a vehicle's payload could serve many more users with more data per user and serve users deeper indoors. Because each vehicle spends much more of its life over the service area serving users and because each of those service hours gathers much more revenue, the average revenue each vehicle can harvest over its lifetime is dramatically higher. That revenue is sufficiently high to pay for the cost and operation of the vehicle, a milestone that would be difficult to reach with a pumpkin balloon. In addition, reasonably sized fleets of these vehicles could reach availability levels of 95% for even small service areas, and for larger areas, it could reach 99% and higher.

Table 8-1 Hammerhead Candidates

Area/Item	***Hammerhead Film-based***	***Hammerhead Fabric-based***
Balloon		
Minimum operating altitude	15.25 km	15.25 km
Maximum operating altitude	20.7 km	18.5 km
Balloon diameter	29.5 m	22.8 m
Balloon length	60 m	57 m
Volume of envelope	26,460 m^3	15,050 m^3
Maximum lifting capability	1875 kg	1340 kg
Propulsion		
Nominal Lateral Power	2300W	1800W
Avg max sustained speed	5.25 m/s	7.8 m/s
Number of props	2	1
Diameter of prop	2.5 m	3 m

Although these 60 m designs were three times as long and six to ten times the volume of Loon's latest and largest pumpkin balloon, they are still relatively small and inexpensive compared to most of the stratospheric airships proposed by industry, which range from 120 to 300 m long. The larger size of those airships is needed to hit the required high availability using propulsion only.

Key Learnings, Innovations, and Challenges

- LEARNING: High availability and profitability are challenging for a non-propelled pumpkin balloon for most markets and communications services.
- INNOVATION: Using lateral propulsion (propellers) with an aerodynamic envelope (airship) could solve both problems when used in conjunction with Loon's altitude-based navigation technology and algorithms.
- LEARNING: To understand the effect of any design variation that affects steering, simulations of the entire system must be performed, including all relevant vehicle characteristics and navigation software behavior, using decades of historical wind data for candidate markets.
- INNOVATION: To accomplish this, Loon developed a detailed, accurate, and highly scalable system simulation toolkit, providing key insights into each vehicle's performance over each potential target market, including availability, GBs served, cost per GB served, and users served. This simulation used decades of historical wind, temperature, and other weather data in conjunction with the production navigation software and algorithms and an accurate model of the candidate vehicle design, as well as an accurate LTE simulator incorporating antenna patterns, sector characteristics, and user and demand density maps of the service areas.
- INNOVATION: To enable a vehicle design optimization algorithm to work through thousands of different designs in a single optimization run, Loon also created an accelerated steering performance grading algorithm that could process new design candidates in a small fraction of the time required for a full system simulation. Combining both tools into our process allowed us to quickly identify strong candidates, refine those into an optimized design and validate the expected performance with a full system and environment simulator.

Airship Design Explorations

After systems engineering analysis demonstrated that such a vehicle was potentially profitable and realizable, Loon began a serious effort to design that vehicle. The initial investigation, code-named Jefferson, provided key learnings, including identifying many key challenges. The Jefferson program then spawned the effort to build this into a commercial product, code-named Hammerhead.

Film vs. Fabric Envelope

Such a superpressure airship is typically designed with a fabric envelope that has multiple layers, such as:

- A gas-impermeable layer that keeps the lift or ballast gas from escaping
- A strength layer, often a weave of strong fibers
- A highly reflective layer, often metalized, that keeps the envelope and internal gases from getting too hot in the daytime and reduces UV degradation

Additional layers are also often present.

Because this fabric is designed to be strong enough to support the worst-case tension without requiring supporting tendons to bear the load, this allows the overall aerodynamic shape of the vehicle to be formed directly by appropriate cutting and seam bonding of the fabric material itself. Another potential benefit of such a fabric is a potentially longer lifespan, with industry claims of three-, seven-, and ten-year flights for airships based on their fabrics. This increase is due to a combination of the inherently stronger material, specific refinement of the fabric for long life, and significantly reduced UV degradation due to the highly reflective outer layer.

Figure 8-2 ***Jefferson fabric envelope testing.***
Taking advantage of -40°C weather in Yellowknife, Canada, to test fabric and envelope strength under pressure.

For these reasons, Loon started a serious effort to co-develop a new fabric and an envelope based on that fabric for use in the next-generation Loon vehicle. Preliminary studies, Computational Fluid Dynamics (CFD) analysis, and wind-tunnel tests showed that a whale shape rather than a teardrop shape was optimal for this environment and speed, with an apparent decrease of drag and increase of stability and controllability.

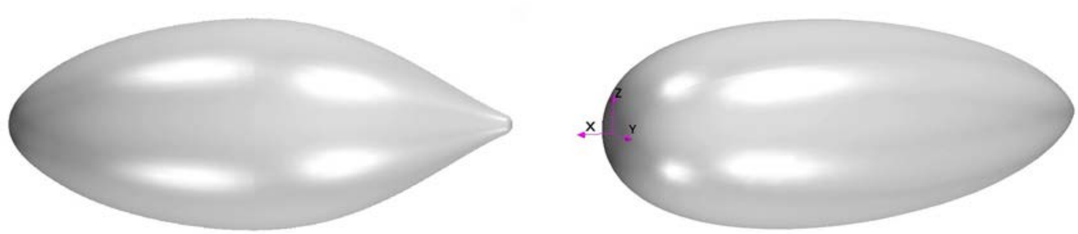


Figure 8-3 ***Whale vs. teardrop shape.***

Despite the clear advantages, there were two concerns about moving to such a fabric. As mentioned above, the fabric weighs significantly more per square meter than the PE film used for Loon's pumpkin balloons, and requires a larger envelope to offset this weight. In addition, Loon's engineering, manufacturing, and operations teams had years of experience optimizing the materials, construction, handling, and operations of PE film-based systems. Therefore, switching was a significant undertaking with a good deal of risk; it wasn't clear how long it would take to regain the same mastery of the new fabric.

Leverage Loon's Film Background

Due to these concerns, Loon simultaneously began developing an aerodynamic envelope shape using the same PE01 polyethylene film that Loon co-developed with a film partner. The basic design started with the same pumpkin shape with diamond-shaped gores attached edge to edge and strengthened by tendons going from the top to the bottom of the balloon, attaching to strong load rings.

Figure 8-4 ***Loon's PE01 pumpkin balloon–starting point for Raptor balloon design.***

Imagine this design then split at its equator, the top half raised, and a cylinder of material added between the two halves of the pumpkin to form a very tall balloon. It is then flipped on its side, becoming a very long sausage shape (code-named Meatloaf), to have a low cross-sectional area in the direction of motion (horizontally).

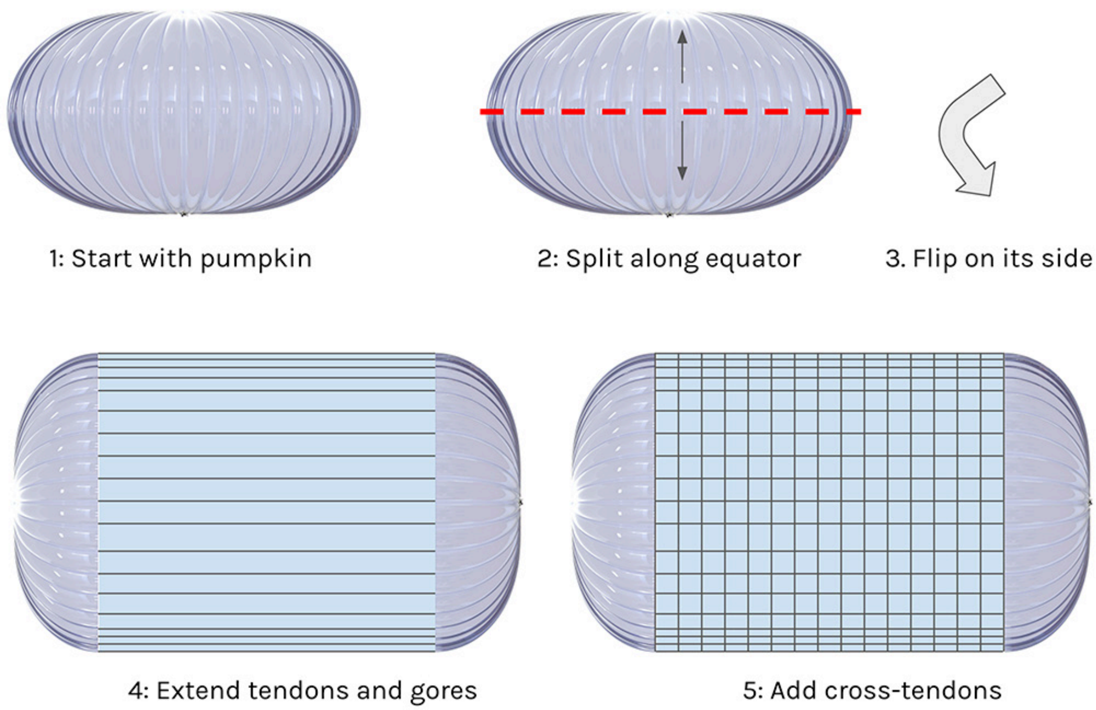


Figure 8-5 Converting a pumpkin balloon design to a Meatloaf.

However, the film itself is not strong enough to directly form this shape, particularly around the cylinder portion. To address this, Loon used additional cross-tendons for the cylinder, which were parallel to the original pumpkin's equator and perpendicular to the existing tendons (which now go from front to back, rather than top to bottom). Doing so provides more strength around that middle, elongated section, where additional tension was required to maintain the shape.

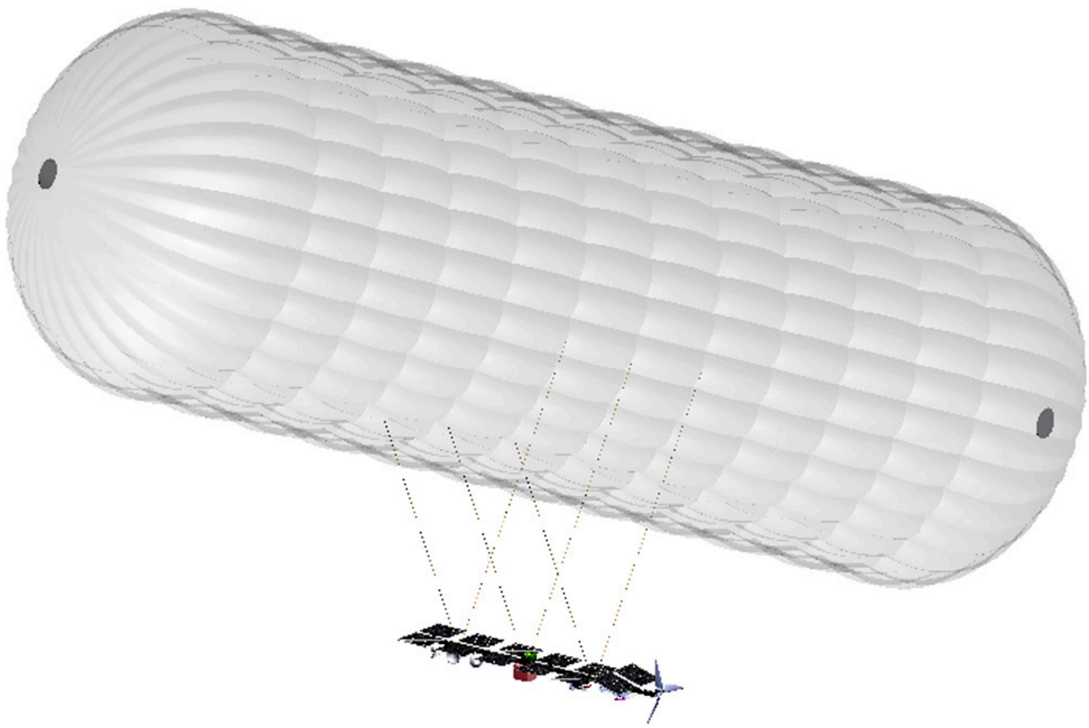


Figure 8-6 ***Meatloaf: Split, flopped, and stretched pumpkin, adding cross-tendons along the way.***

With this technique, Loon engineers could attain a shape with significantly lower drag than a straight pumpkin or flipped prolate pumpkin (tall, skinny pumpkin flipped on its side). However, the drag was still far higher than what could be achieved with the fabric because the overall shape was more sausage than whale or teardrop, and the many bumps formed by the grid of tendons and cross-tendons disrupted the airflow.

Both issues were significantly reduced in later prototypes through the following refinements:

1. First, we changed the gore shapes to taper over a longer portion and more quickly toward the front and back of the vehicle, forming a more football shape. This showed a clear but insufficient decrease in drag.

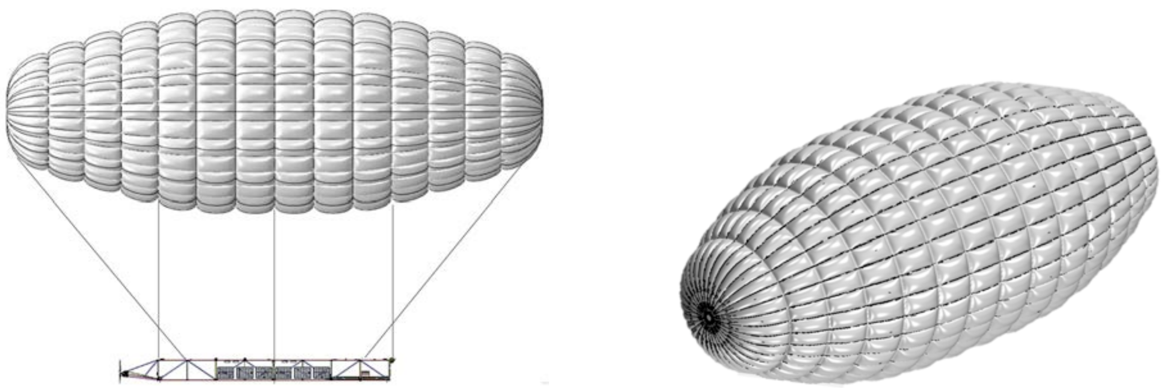


Figure 8-7 Bumpy football: front/back symmetrical.

2. Then we changed the gore profiles again to taper more quickly in front (direction of travel), and less quickly in back, allowing the overall shape to resemble the desired teardrop or whale shape more closely. Again, drag improved but not as much as desired.

Figure 8-8 Bumpy Teardrop: asymmetrical.

3. Finally, we designed a non-tendoned PE01 aeroshell to cover the cross-tendoned envelope, covering up and smoothing out most of the bumps.

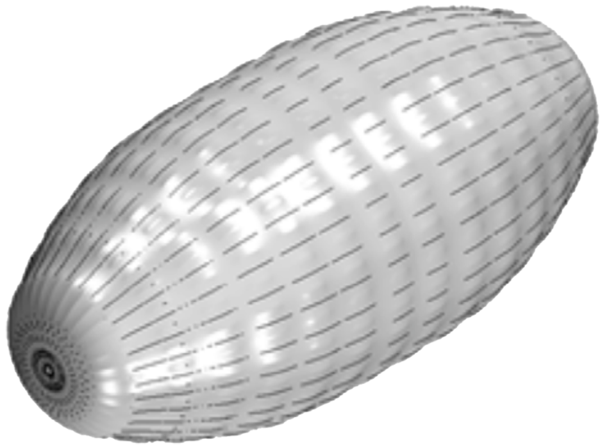


Figure 8-9 ***Mamba: bumpy teardrop with skin/aeroshell to reduce bumps.*** (bumpy teardrop not shown)

Combining these refinements resulted in a very reasonable drag: still not as low as the fabric-based envelope designs, but much less expensive and with a much better mass efficiency (mass lift capability versus mass of lifting body), which resulted in a smaller envelope than fabric could deliver.

Figure 8-10 ***Some of Loon's balloon design team standing inside a Bumpy Football balloon.***

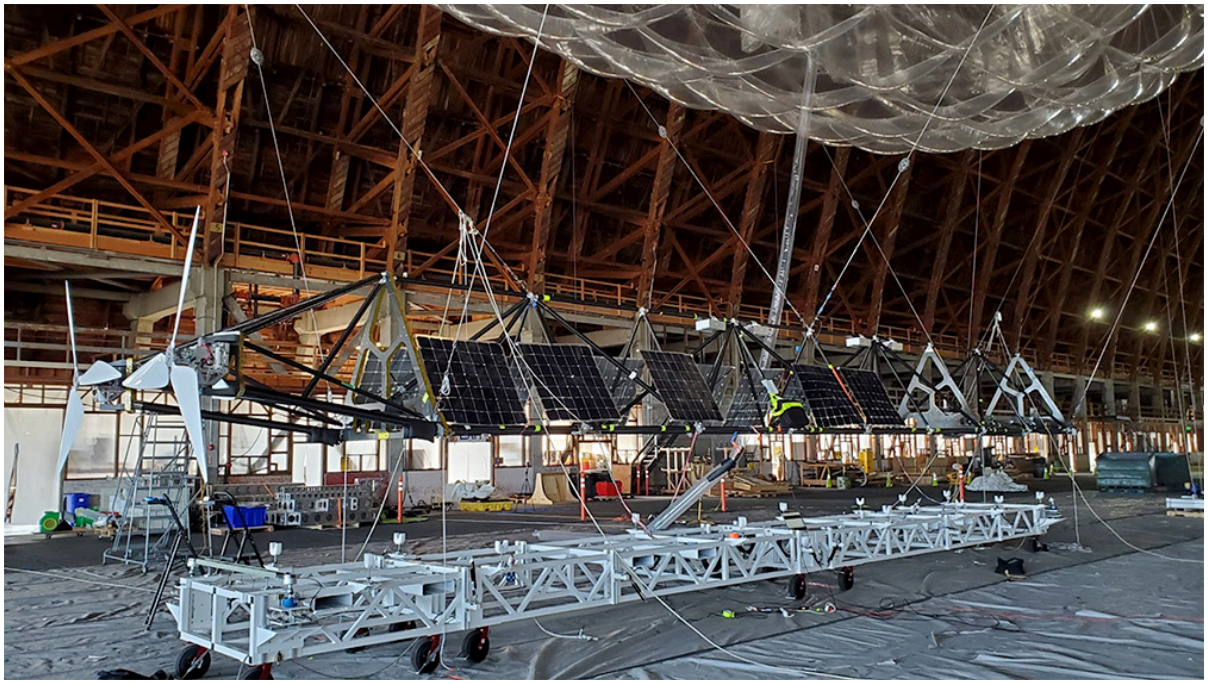


Figure 8-11 ***Alphadon prototype bus with propulsion prototype undergoing tethered launch trials before test flight.***

Figure 8-12 ***Bumpy Football balloon and Alphadon bus prototype in tether launch trials for Hard Launch process.***

Key Learnings, Innovations, and Challenges

- LEARNING: Nearly all proposed stratospheric airships use expensive, strong, heavy multi-layer fabric and, due to the weight, a much larger envelope would be needed for the same lifting capacity versus a PE01-based pumpkin balloon. Loon decided to concurrently design two aerodynamic envelopes. The first generation was expected to be based on the same advanced transparent PE01 film Loon developed for pumpkin balloons. Later generations were expected to use the multi-layer fabric above after the cost and characteristics of that fabric reached the target level.
    - » Advantage of fabric: stronger, more aerodynamic shapes are possible, lifetime (may) be longer. In operation, these vehicles take more advantage of propulsion due to their lower drag but do still benefit greatly from altitude-based steering.
    - » Advantage of PE01 film: much lighter and cheaper. The Loon team has mastered the use of the film across many design and operational aspects. These vehicles will take more advantage of altitude-based steering but benefit greatly from propulsion as well.
- CHALLENGE: Such an airship is much larger than Loon's pumpkin balloons. This required a company-wide commitment as this change in direction had a major impact on all aspects of designing, building, launching, and operating.
- INNOVATION: An aerodynamic envelope can be made from a single-layer polyethylene film (at least if it's PE01), but it's non-trivial and would require further refinement. Loon wasn't done, but the best solution we found:
    - » Use basic pumpkin balloon construction plus cross tendons to form a teardrop or whale-shaped envelope.
    - » Use thin PE01 skin (aeroshell) to reduce the impact of cross-tendon-induced bumps.
    - » A length about 2.5x to 3x the diameter is a good compromise between drag and mass/lifting efficiency. A shorter design weighs less for the same lift, but longer has less drag. A full trade study on this was not completed.
- LEARNING: Such a PE film-based vehicle still has worse drag (more power per speed) than a whale-shaped fabric airship, but the difference in cost appears to make this worthwhile for many services across some markets.
- LEARNING: Wind tunnel tests show that, even at these relatively low speeds, the amount of drag is sensitive to small variations in shape and surface texture.

Airship Launch

One innovation that enabled Loon to reach high launch rates of pumpkin balloons was the design and construction of several large launch cranes. In addition to largely automating the launch process, these also protected the balloon from the wind while it was being filled with helium for launch (which could take over 20 minutes). This reduction in sensitivity to wind speeds significantly expanded the number of launch days available.

Like others, Loon launched the pumpkin balloons immediately after filling the amount of helium required to ascend, reach, and remain at the proper float altitude range. This meant that about 95% of the total balloon volume was empty, leading to the classic drooping balloon bag shape on launch, being lifted by a small bubble of helium at the top.

Loon initially assumed that we would launch the airship this same way, which we started calling soft launch, but that led to many complications around the ascent process. Unlike a radially symmetric pumpkin with a compact bus/payload attached at a single point, the airship has a much larger and very asymmetrical balloon, two internal ballonets and a long and skinny bus/payload, which is hanging from a dozen attachment points across the length and width of that balloon. At launch, all this would be hanging from the helium bubbles in the two separate ballonets, such that the layers of materials, tendons, attachment cables, propellers, and more are all hanging, clumped together below the sagging envelope. It is much more difficult to enable a repeatable and reliable soft launch and ascent of this than for the pumpkin.

Figure 8-13 ***Bumpy Football and Alphadon bus with propulsion undergoing Soft Launch tethered testing.***

Figure 8-14 Vee Launch fill process as part of Soft Launch.

Due to the complexity and risk of the soft launch approach, Loon began to consider a fully inflated launch process instead. In this case, the entire air ballast chamber would be filled with sufficient air for the envelope to be taut.

The advantages of the fully inflated launch are clear:

- The balloon is filled, shaped, and taut before launch, tensioning the guy lines supporting and stabilizing the bus/payload and keeping the many pieces adequately separated to avoid damage.
- A complicated unraveling process and associated mechanisms do not need to be designed to accommodate the transitions that the balloon+bus combination would go through during the ascent.
- This expensive balloon can be pressure tested before launch to ensure it and its expensive bus and payload will make it to float altitude.

But the disadvantages are significant:

- The air inside the envelope will encounter increasingly cold temperatures on the ascent, causing condensation and then freezing of that condensation throughout the volume, either requiring filling with extremely low humidity air or potentially impacting the reliability of anything in contact with that water and ice.
- Filling such a large volume takes a long time, significantly decreasing the launch rates.
- Over the course of the ascent, 95% or more of that air needs to be vented to avoid over pressuring the envelope, starting almost immediately after launch. The exhaust valve supporting this needs to be very large to support the high airflow needed for the planned ascent rates. This valve and exhaust process needs to be extremely reliable during the ascent, despite the condensation and freezing concerns, and needs to stay firmly closed for the remainder the vehicle's lifetime.
- Leaving such a large vehicle on the tarmac during a long fill and prep cycle could subject it to winds, blown debris, temperatures, and humidity for which it would not otherwise need to be designed. It is safer to complete the launch prep inside a protected hangar.

Fortunately, for Hammerhead, we assumed that we would not need to reach the same launch rate that was initially expected for Loon (tens of thousands per year) for two main reasons:

- The required overprovisioning of these fleets was expected to be much smaller, reducing vehicle counts significantly.
- The vehicles themselves had greater than one-year projected lifetimes versus the three-month forecasts for early Loon.

So, this reduces some but not all the concerns.

Loon had a series of experimental launches planned to work through these concerns, and, at the time of wind-down, no definite winner had been declared.

Airship Descent

The airship creates a few new challenges that must be considered when designing the descent process. Compared to the various termination and descent approaches prototyped and productionized for Loon's pumpkin balloons, the airship required new thinking to identify the viable options and the advantages and disadvantages of each.

Relative to pumpkin balloons, it's easier to control the start-of-descent location with airships but harder to predict how far they'll drift on their descent.

For the pumpkin balloons, the primary factor affecting the actual landing location was the poor steering control as the vehicle approached the termination (start-of-descent) location. However, pumpkin balloons proved to be quite predictable in their descent profile since the Flight Termination System (FTS) cutters, and vent holes were at the top of the balloon during descent, so lift gas was vented continuously at a predictable rate.

With the much larger airship, there is a corresponding increase in the importance of landing accurately. However, that much larger vehicle would also feature significantly improved steering capability, allowing it to reliably get much closer to the optimal termination location than was achievable with Loon's pumpkin balloons. At that point, the descent profile may be significantly more difficult to predict. Increasing the accuracy of that landing prediction is one of the primary objectives of the termination and descent system design. Various approaches were considered.

Traditional Loon Approach

Taking a traditional Loon venting and parachute descent approach presents some new challenges given the elongated configuration of the airship envelope and the location of the FTS cutters:

- It is difficult to ensure venting continues at predictable rates because the venting hole is no longer at a single reliably maintained position at the top of the deflating balloon. On the airship, the two assemblies that house the FTS cutters are now at the front and back ends of the envelope, which will begin to sag as the pressure drops, and the envelope begins to lose its shape. This can trap significant and unknown amounts of lift gas. Even if a venting location is added at the initial top of the shaped envelope to avoid trapping gas, the envelope will still distort and potentially trap an unknown amount of gas, which makes the buoyancy of the airship difficult to estimate and affects the prediction of the descent profile.

- As the envelope distorts, it can take on less predictable shapes for modeling drag. A venting pumpkin balloon has a very consistent teardrop shape as gas volume decreases. As a large horizontal shape loses significant gas volume, it can take on different shapes leading to different drag profiles.

Both factors could create errors in the forecasted drift and increase the landing zone prediction error.

Steerable Parachutes

Steerable parachutes can potentially add a great deal of landing zone accuracy. By tracking the descent progress, the descent steering algorithms could adjust on the fly to keep the vehicle heading for the landing zone and even reduce the impact of the lack of predictability of descent rate or winds along the descent path.

The degree of descent steerability is very dependent on the achievable glide ratio. With Hammerhead, the large, unruly combination of the deflated balloon, the bus/payload, and the steerable parachute would degrade the glide ratio significantly, and its instability will degrade it unpredictably. Despite this, a steerable parachute may still improve the overall accuracy enough to make it worthwhile. It also improves the ability to steer away from specific undesirable locations near the landing zone, should the initial prediction be poor.

Bus Line Termination

In this method, a set of suspension lines would be cut during termination, allowing one end of the envelope to rise with buoyancy. That would maintain its ability to vent continuously and provide a repeatable vent rate for altitude vs. time modeling for descent trajectory simulation. Note that in this method, the lateral-shaped envelope has transitioned into the well-understood venting and descent configuration of pumpkin-style flight systems like v1.4 and v1.6.

Separation during descent could be developed as in v1.6 or landing unseparated could be developed as in v1.4.

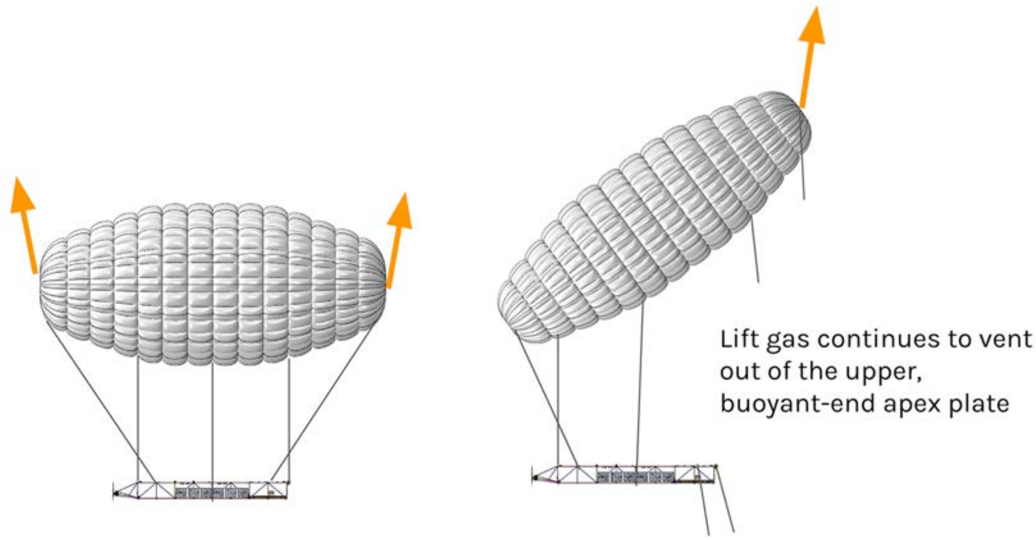


Figure 8-15 ***Bus line termination descent method allowing one apex plate to remain at the top of the lift gas bubble (envelope distortion not depicted).***

One risk is that, as the envelope distorts and the loose suspension lines flop around, there is a high likelihood of those lines or the envelope entangling with the bus and service payload, risking breaking components off or requiring impractical and mass-prohibitive hardening of all components. Variations to address this issue were also considered.

Rigid-Envelope Controlled Flight to the Ground

As an alternative to Loon's traditional venting and parachute descent method, the flight vehicle could theoretically be kept intact, with its shape maintained even as lift gas is vented and air ballast is introduced to maintain the total volume.

This would keep the flight system in a not-quite-buoyant state but, more importantly, a structurally stable state until landing. Due to the large volume of air to be added and managed as atmospheric pressure increased, tremendous energy and time are required to run the ACS, which would significantly slow the descent, potentially to several days.

Although this strategy for landing is used by commercial passenger and military airships and initially seems attractive, there are two significant differences between those vehicles and Hammerhead:

- The pressure ratio between the ground and the operating altitude: For most passenger airships, this ratio is about 1.5:1 (although proposed designs approach 2:1), but for a stratospheric airship, it is about 20:1. This affects many aspects of the design and operation, including requiring the balloon to be at least 10x as large to lift the same weight.
- The airspeed capability at landing: For typical airships, this is 60 to 100 mph (100 to 160 km/h), but for Hammerhead, it would likely be less than 6 mph (10 km/h) as the prop and motor would be designed for the lower density air in the stratosphere and unable to operate at the higher torques required for higher speeds near the ground. Increasing the power and torque capability of the propeller to compensate would decrease its efficiency during normal operation. This also does not address the much higher energy required for more than short bursts of that improved speed.

With such a large envelope surface area and the quite common potential for wind speeds that exceed the vehicle's airspeed at lower altitudes, Hammerhead's propulsion would not be designed to support the thrust required to make headway against the winds nor provide much steering ability. Because the vehicle is descending so much more slowly than the approaches described above, unpredicted wind directions or speeds could blow the vehicle far away from the intended landing zone.

Even a modest wind could drag the flight vehicle across the terrain as the vehicle approaches the ground. Preventing this with a passenger airship-style anchoring tower would require the propulsion to be sufficient to reliably approach such a tower, which it is not, as described above.

These issues lead to large landing zone prediction errors due to the extremely long descent duration and the large drift distances. See "Landing Zone Prediction Accuracy" on page 137 for more context.

Note: All the descent work described above was by analysis and from experience with descents of pumpkin-style balloons. The first prototypes of shaped balloons with suspended long buses were being readied to trial some of these methods. They are written as ideas being considered rather than necessarily useful implementations.

Key Learnings and Challenges

- CHALLENGE: Landing such a large, complicated, and asymmetrical airship presents several new challenges and will need to be studied in depth.
- LEARNING: Loon's traditional approach is a good place to start due to the fast descent time. Hammerhead's high steering accuracy can nail the optimum termination (start-of-descent) location making up for the likely less accurate descent path prediction.
- LEARNING: Although maintaining full volume and stable shape while descending sounds attractive, the massive envelope size and much higher air density near the ground significantly increase the descent duration and the drift during that descent, resulting in a corresponding increased landing zone target. An optimum Hammerhead propulsion system is largely ineffective against the near-ground-level winds, requiring significant over-build of this system and increased energy requirements.
- LEARNING: Steerable parachutes are a potential win but need much more study.

High-Capacity Future Access System

In addition to the airship itself, Loon was designing its Future Access System (FAS), a combined LTE+5G base station with new features and objectives:

- Much higher capacity versus Loon's LTE-Gen1 and Gen2 systems
- Use of commercial off-the-shelf boards wherever feasible to minimize Loon engineering work required and to reduce time-to-market for new advances
- Lower weight and power per GB served, achieved by using many sector systems designed for low power but also through new antenna and RF designs
- Provide both a coverage system and a high-capacity system:
    - » Coverage would be similar to the fixed direction four-sector LTE-Gen2 system (Rickenbacker) but could be extended to up to seven sectors if stronger signal levels were needed
    - » The high-capacity system would consist of three or four multi-sector steerable gimbals, each with four or seven sectors
- Tentative architecture and key parameters (including the number of sectors, beams, gimbals, antenna gain, and transmit power) were determined as part of a full-vehicle Hammerhead systems engineering optimization exercise

LTE Sizing in the Full-System Optimization

The full-vehicle system optimization had a dual objective:

- Minimize cost per GB served and
- Maximize availability

Increasing the mass and power consumption of the LTE system has a direct and sizable impact on the revenue that the system can earn but also increases the power needs, which in turn drive up the cost of the vehicle. This is a classic systems engineering problem:

> *How much mass and power should be allocated to the LTE system to minimize cost per GB_served, and availability?*
>
> *And, for a given mass and power, what is the best architecture and sizing for that LTE system?*

As described in "Systems Engineering Design Approach" on page 42, Loon built an extensive systems engineering optimization toolkit to answer questions like this. This toolkit was necessary because of the system's complicated, highly interdependent nature and sensitivity to wind and weather patterns, population density and adoption rates across the service area.

For the past year or so, Loon's engineering team developed and modified these tools to determine what size, shape, and architecture the airships should be and what payload the airships should carry. This exercise pointed Loon toward the 60 m first-generation airships described earlier and showed that the optimal LTE architecture should carry between 16 and 25 sectors in the combined Coverage system plus multi-gimbal Capacity system configuration.

Many Small Sectors

The three or four high-capacity gimbals, each with four to seven sectors, could be aimed towards population clusters or towns to keep those towns covered even as the airship drifts overhead. The sectors were each expected to cover a circle 2 to 4 km across if pointed directly down, or an ellipse about twice that area if pointed at 45° (halfway to the horizon).

These small sectors have several advantages:

- Smaller sectors distribute energy over a smaller area, resulting in stronger signal levels for the users in that area, increasing the users' data rates and ability to receive service indoors or in cars.
- Smaller sectors also allow a much higher density of users (greater users per km^2) because the number of square kms decreases and the signal level improves, as described above.
- The ground size of these sectors is small when pointed straight down but grows as the gimbal is pointed closer to the horizon. Because of the high gain, at 60 km from the balloon position, the signal strength remains strong enough to support in-car use by low-end phone users.

But small sectors have one clear disadvantage: the area covered is much smaller, leaving larger areas uncovered. Even with three of the seven-sector gimbals, the significant majority of the balloon's footprint was uncovered.

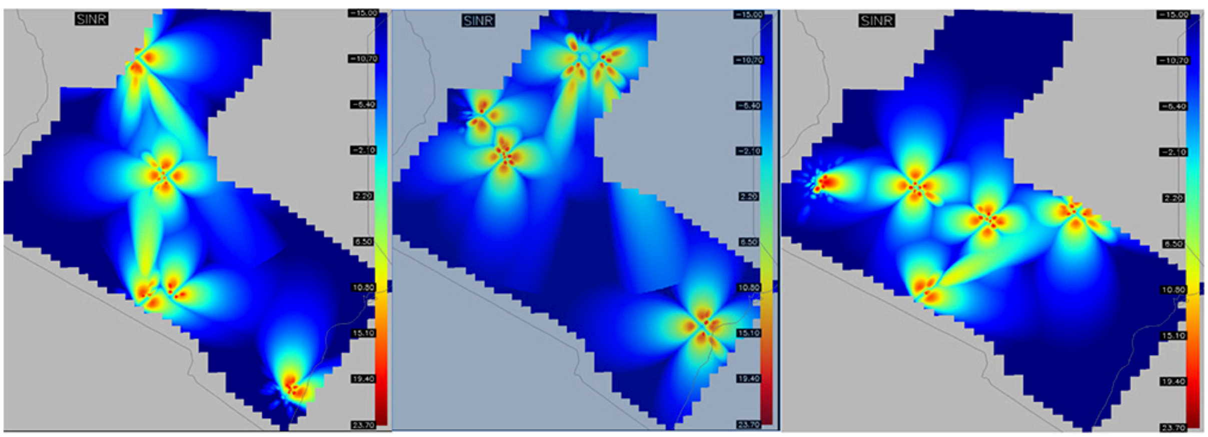


Figure 8-16 ***Simulation of a new high-capacity payload design showing steerable multi-sector beams and their coverage pattern on flights over Kenya.***

The relatively small coverage of these high-gain sectors was also the incentive to include the LTE-Gen2 Rickenbacker-like four coverage sectors. However, their coverage advantage is not as high as it might first appear. As described in "Service Payload Architecture" on page 178, only outdoor users with high-end smartphones were served out to the 50 km radius. Users with less expensive phones more commonly found in Loon's service area would only receive outdoor coverage over about 1/4th of that footprint, to about 25 km radius from the balloon position.

Antenna and Sector Forming

Digital beamforming was considered for the control of the individual sectors within a steerable multi-sector group but was also considered for steering an entire group. Such a beamforming approach has significant advantages where extreme or complicated shaping of the sectors is needed, but this approach comes with significant power needs. In general, the power premium for digital beamforming can be minimized by reducing the ratio of the number of antenna elements to the number of sectors formed by those elements. That ratio, which is very dependent on the carrier frequency (band), the desired size and shape of the sectors and related factors is often 4x to 20x, which has a large impact on power consumption. However, for many applications, that's a worthwhile trade because of the continuously flexible and detailed control over the antenna patterns. Note that most of the power premium of digital beamforming results from the many RF chains required, each with its own transceiver and other electronics, even if the transmit power for each RF chain was lower than the alternative.

Loon had investigated several options for power reduction of digital beamforming, but none met our need to maximize GBs delivered with the least mass and power. This fact pushed us away from digital beamforming and toward an architecture with one antenna feed per sector, using a very lightweight reflector to create the gain and pattern. Loon decided on a shared-aperture/shared-reflector for each gimbal to minimize the mass for those reflectors and their supporting and steering mechanisms. Several options were being examined, and, at the time of the company's wind-down, the team was working toward an offset, shaped parabolic reflector, with multiple off-focus feeds per reflector.

The two previous generations of Loon's LTE system used Photon Harvester heat sinks to dissipate heat during operation and help keep the boards from getting too cold at night when not operating (read more about thermal management in "Thermal" on page 152). The disadvantage of this approach was the still significant heater power required on the colder nights, which became a sizable portion of the overall nighttime power consumption, thereby limiting ACS maneuvering. Because the photon harvester is basically a linear, single-mode system, as the operating heat dissipation requirements continued to increase, so did the heater power.

With the FAS, Loon was working on a more traditional forced-air convection system, designing a single thermally controlled cabinet to hold all the LTE baseband and the CommsNode electronics. This solution would share the insulation weight across those boards and allow operating boards to warm up neighboring powered-down boards. This approach, combined with the plan to keep at least some LTE sectors serving at night, led to an architecture that would waste little power in heating electronics but was mass and power-efficient in cooling during high-powered operation.

Key Learnings, Innovations, and Challenges

- INNOVATION: Detailed and accurate LTE (and 5G) payload models are needed to do good model-based systems engineering trade studies and system optimization studies to adequately consider the costs and benefits of the wide range of architectures and sizings.
- LEARNING: Digital beamforming's additional flexibility is valuable for maximizing service while minimizing interference but comes at a high cost in power and mass. For Loon's target markets and service, it looks like the best approach is multiple 2D mechanical gimbals, each with a 7-sector shared aperture reflector with individual feeds per sector. A shared aperture reduces the mass of the reflector. Individual feeds (rather than a phased array or digital beamforming) are the lowest electronics power and mass per sector.
- INNOVATION: Common baseband based on COTS terrestrial boards with thermal assistance. Not as power efficient as desired but dramatically less engineering work.
- LEARNING: Average power consumption, not peak, is most important since total energy consumed was the priority. Investigating and implementing off-peak power reduction techniques is required, not just because of off-peak hours but also because of off-peak sectors (those pointing toward lower user density areas).
- LEARNING: The photon harvester, though brilliant and effective for low-power electronics, is inefficient for higher-power systems, requiring a high heater power to keep the boards above -40°C on cold nights. Instead, Loon needed to move to a more traditional forced-air convection approach using a thermally insulated chassis containing multiple PCBAs. This would minimize heater energy required when not operating. Custom fans would be required; standard fans cannot handle the low air density nor extremely low temps.

Airship Next Steps

Although Loon had years of experience with stratospheric superpressure balloons, it was understood that Hammerhead would be a major undertaking. The estimate (in mid-2020) was that it would be a few years before it would be ready for pre-commercial trials. Such an effort would require an increased engineering effort to design the all-new balloon, bus, payload, and launch systems and increased operational effort to support the more extensive supply chain and manufacturing.

We had sketched out two major size steps with several dozen prototype test flights, each targeted at specific learnings or decreasing specific risk areas. Because all major subsystems were being redesigned simultaneously, there was, understandably, a reasonable amount of risk of schedule slip. Some of those concerns were due to the significant size increase, requiring an alternative manufacturing, assembly, and launch process. However, many companies have built and launched airships that were four to ten times the size of the proposed Hammerhead vehicles, so there was confidence that we would make this transition.

Preparations for the Transition to Hydrogen Operations

Whether Loon transitioned to an airship or continued with the pumpkin architecture, it was clear from early in Loon's history that we would need to shift operations from using helium as the lift gas to using hydrogen. It was a question of when not if. Multiple elements are involved in the timing, but a significant one was waiting until the flight vehicle, launch site, and systems design were stable so that extensive validation testing of safety-critical hydrogen functionality would be valid for a long-term production run.

Loon planned to switch to hydrogen for two main benefits: reduced cost and removing reliance on a non-renewable resource. Helium is expensive to collect from natural deposits and has a rapidly fluctuating price. It is also a limited non-renewable resource that is being gradually lost; once helium is vented to the atmosphere, it will float up and eventually escape the atmosphere entirely. Because helium is necessary for many scientific, medical, and commercial processes, it makes sense to use alternatives wherever possible. Although hydrogen weighs about half as much as helium, it only brings a marginal increase in lifting capacity for a given vehicle because the key factor is the weight compared to the air that it is displacing.

Loon studied extensively what would need to be done to prepare for hydrogen operations but did not develop specific implementations since it was clear that the flight system and launch infrastructure would be evolving for some time before stabilizing. Therefore, implementation details would need to be updated once they had stabilized and final systems could be tested.

The importance of very carefully designed implementation follows from hydrogen's very wide range of flammability. When mixed with air, hydrogen is flammable at a range of 4% to 75% concentrations and explosive from 18% to 59%. Significant safety measures, equipment, and procedures are needed to operate with it safely. Considerations include launch site facilities, equipment, and process, as well as flight vehicle equipment and procedures.

Launch Facilities

Facility design is informed primarily around maintaining sufficient standoff distances from lines containing hydrogen and zones in which hydrogen hazards can be present. While large-scale fire or explosion are two primary hazards, lesser ones such as leaks, both burning and not, are important to consider since hydrogen burns with a nearly invisible flame and is odorless.

Key codes and references relevant to the implementation of hydrogen systems include, but are not limited to, NFPA 55, NFPA 70, 29 CFR 1910.119, CGA G-5.4 and CGA G-5.5, ICC Performance Code Chapter 22, ICC Fuel Gas Code, and the International Fire Code.

Hazard zones and distances are established either through regulatory requirements or through analysis and characterization testing of certain events. For Loon, a worst-case hazard includes the large balloon envelope filled with hydrogen where somehow air gets introduced to create a flammable mixture of large volume. The thin plastic nature of the balloon envelope film made this something to be considered carefully. This was not a typical situation considered in codes.

Two modes of flame propagation exist relevant to hydrogen, with vastly different effects: detonation and deflagration. These are important to consider in assessing hazard zones.

- Detonation is a type of combustion involving a supersonic exothermic front accelerating through a medium that eventually drives a shock wave propagating in front of it.
- Deflagration involves a subsonic exothermic front propagating through heat transfer; hot burning material heats the next layer of cold material and ignites it. Relative to detonation, this is more than an order of magnitude less overpressure but with higher thermal radiation.

Loon partnered with research institutes to conduct small- and full-scale live testing of these hazards and detailed analysis of potential events. This work aimed to establish the scale of hazard zones given the conditions of hydrogen volume, stoichiometry (mixture with air), and containment, including far-field structures in the path of the overpressure front like the launch crane structure itself. Both analysis and testing indicated that both scale and obstruction/containment were key factors relevant to deflagration to detonation transition. Still, there was no truly deterministic way to be certain an event would not transition.

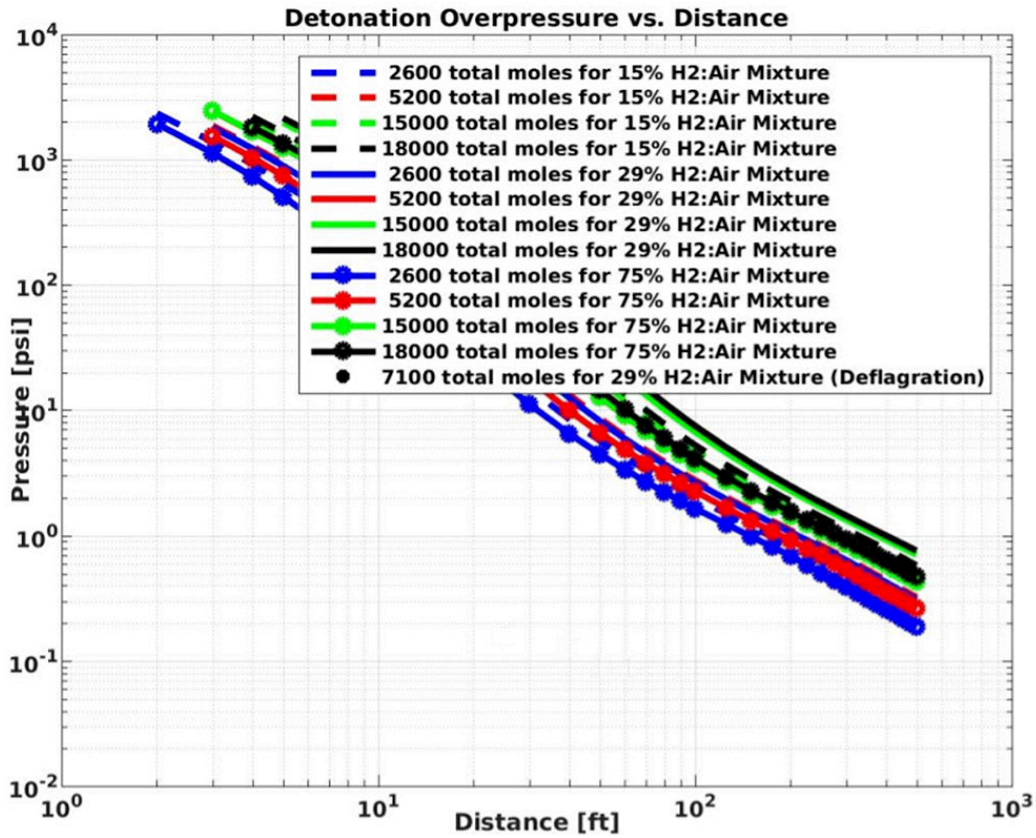


Figure 8-17 ***Fill sizes and hydrogen-air mixtures with resulting overpressures and distances for detonation conditions from a preliminary study for a specific proposed configuration only.***

Given the uncertainty that ignitions would stay in the deflagration regime, Loon considered a low-risk approach as assuming worst-case conditions for air mixtures with detonation and then developing hazard zones from that. The analysis assumed the full TNT-equivalent potential of any particular balloon volume and hazard zones were determined based on an overpressure limit criterion. This limit could pertain to effects on exposed personnel (e.g., knockdown and eardrum damage) and damage to structures or equipment (e.g., unreinforced glass breaking and cinder block cracking).

By establishing the overpressure effect criteria and determining worst-case ignition characteristics, and then applying a safety factor, one could determine standoff distances that were safe for all conditions.

The effect of this process on launch site layout would be to include large standoff distances and robust remotely operated filing equipment.

Around the launch site, hydrogen flame detectors, which can detect the presence of nearly invisible hydrogen flames, could cover every hydrogen-filled line. Remote turn on, turn off, and purging and venting equipment allow for putting all lines into a safe condition before personnel can enter the hazard zones.

If a launch aborts after the fill has completed and the fill tube has been crimped closed, the hydrogen already filled must be evacuated. To vent the hydrogen already filled, the Flight Termination System (FTS) could be triggered on the top plate to cut a hole in the lift gas chamber. The buoyancy of the hydrogen combined with the weight of the envelope would ensure all the hydrogen is vented, removing the hazard.

Flight Vehicle Equipment

All equipment on the flight vehicle that could interact with hydrogen needed to be developed with this in mind.

Key balloon systems that were studied for hydrogen compatibility included:

- The Apex FTS PCBA, because it was mounted right above the FTS cutter windows (so it would sit within the path of venting hydrogen) and contained high energy supercapacitors to fire the FTS squibs.
- The altitude control system (ACS) compressor and valve motor. Although not normally in contact with the lift gas, in the event of a ballonet leak or damage, hydrogen could leak from the lift gas chamber to the air ballast chamber, which could then contain a flammable mixture of hydrogen and air, which could be near the ACS during operation.

The Apex FTS board contained supercapacitors to provide power for the squibs actuating the FTS. These supercapacitors were overmolded to prevent hydrogen mixture contact, and this approach tested live for effectiveness. The squib cutters themselves were also live tested to ensure they could not ignite a flammable mixture.

The ACS contained a powerful motor to drive the compressor and a smaller one to drive the valve. Mitigations were put in place to prevent arcing and to seal high power components and this system was tested live for effectiveness.

Descent and Abatement

A crucial element of safe operations would be ensuring that all hazardous amounts of hydrogen were vented from the balloon during descent and before landing. Since various sources of ground-based ignition were possible, including power lines, vehicles, and cigarettes, it was key that the descent incorporated a hydrogen abatement system to prevent a flammable mixture from remaining in the balloon on the ground.

Two proposed flight termination mechanisms also operated as abatement systems, one as a primary and one as a secondary backup. The primary system allowed the bus to drop 20 m on a decelerating tether while pulling a gore-cutting cord through the side of the balloon, thereby creating a large opening, approximately five meters long. The secondary backup system was a line pulling mechanism that would pull a cutter back up to the top plate and would create a cut of approximately two meters at the top of the balloon.

Both systems were developed before Loon's use of the reverse ballonet architecture and so would need to be adapted or replaced as needed depending on the balloon design at the time of transitioning to hydrogen, i.e., pumpkin or airship.

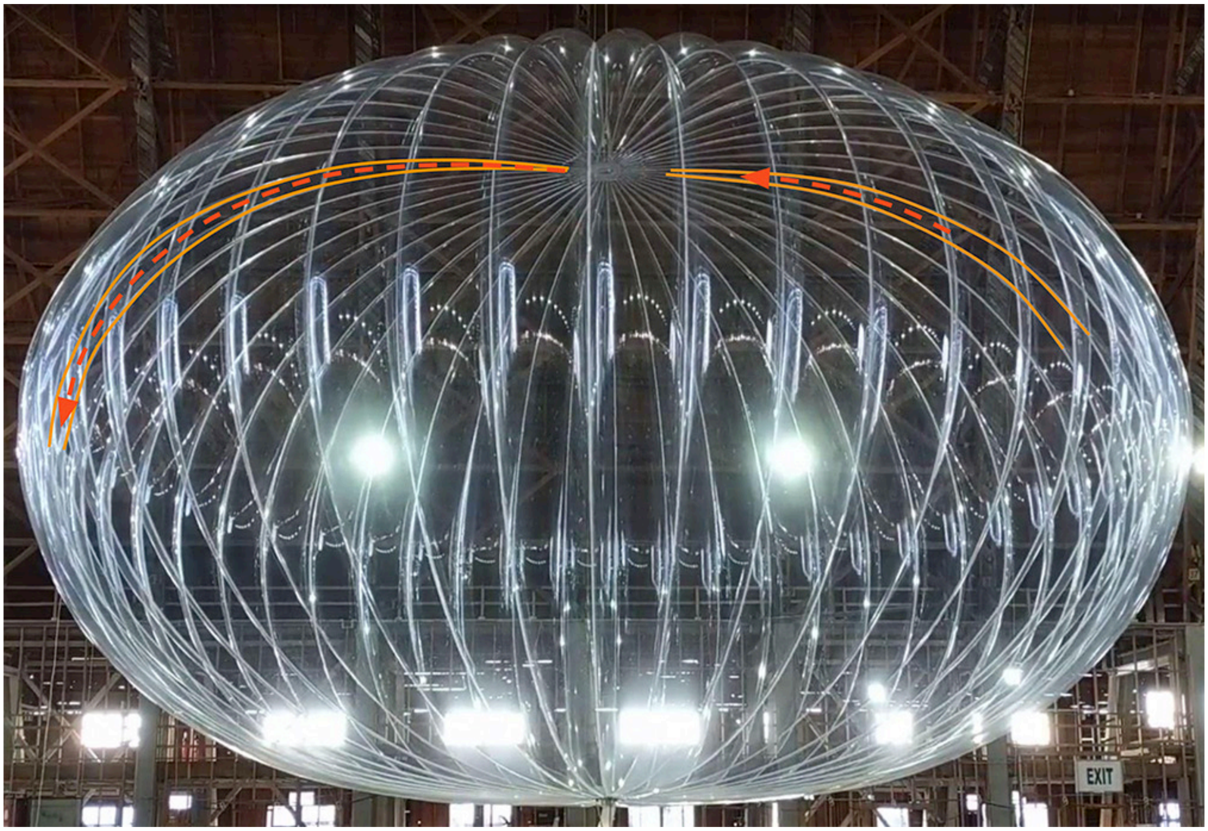


Figure 8-18 ***Abatement mechanisms cutting paths in dashed red arrows, gore cutting cord (left), line pulling cutter (right).***

To test the effectiveness of this dual abatement system, a flight test campaign utilizing both the primary and secondary cutters was conducted using helium, and any remaining lift gas pockets on the ground were measured and sampled for lift gas concentration. This process was meant to build confidence that, in all descents, there would be no hazardous level of hydrogen remaining in the balloon on the ground.

With this level of study and preparation, Loon could monitor the development of the commercial vehicle and launch systems, understanding what would be needed to develop for a specific implementation once those systems stabilized and were ready for scaling. For any company looking at developing a balloon operation utilizing hydrogen, a key lesson is to do the research and work required to develop a detailed implementation plan appropriate for the systems and operation involved. What is described above is only Loon's study of some of its needs to prepare for hydrogen transition. It is not intended as specific implementation guidance.

Dynamic Spectrum Strategy

To manage the topology and network orchestration of its complex mesh network and to avoid interference with incumbent and priority spectrum users in E-Band, Loon designed and deployed Minkowski, a temporospatial software-defined networking (TS-SDN) system. Minkowski creates a model of the spectrum environment where Loon is operating to ensure that Loon's network does not interfere with (or receive interference from) existing spectrum users. To create the model, Loon relied on publicly accessible databases of spectrum licensing and users in E-Band, such as Comsearch's E-Band database in the US.

However, only a small number of national spectrum regulators have created such databases. Where they do exist, they tend to be limited to terrestrial, fixed point-to-point service links. The lack of databases makes it more difficult for non-terrestrial networks to avoid interference, contributes to increased costs for network planning and deployment, and limits the efficiency of spectrum licensing. This issue will become increasingly important as more networks deploy in E-Band, from 5G backhaul to non-terrestrial networks (e.g., HAPS, NGSO satellites, and aeronautical networks).

Increased congestion within E-Band was anticipated, and increasing numbers of networks with moving nodes will need to be coordinated. Loon, both by itself and with the Dynamic Spectrum Alliance (of which it was a member), advocated for countries to adopt database-assisted light-licensing frameworks that could accommodate both existing and emerging use cases in E-Band. This advocacy had multiple components:

- Creation of a comprehensive database of E-Band links in line with the U.S. Comsearch database through an audit of existing licenses.
- Making the database publicly accessible, consistent with confidentiality and national security obligations.
- Enabling real-time updates to the database, including modeling of antennas in motion. The US FCC refers to this manner of spectrum management as 3D Spectrum Management.

An example of Loon's advocacy on this issue may be found here: Modernizing and Expanding Access to the 70/80/90 GHz Bands.[1] Loon encourages all interested parties to continue such advocacy to better enable similar advanced use cases and more efficiently utilize the available spectrum.

1 https://ecfsapi.fcc.gov/file/10806701710599/FCC%20WT%2020-133%2C%2070_80_90%20GHz%20(Aeronet)%20-%20Loon%20Comments%20(AS%20SUBMITTED).pdf

Collaborative Traffic Management in the Stratosphere (CTMS)

Like low-altitude Unmanned Aircraft System (UAS) operations, high-altitude operations, particularly at the scale and with the dynamic nature of automated fleet systems like Loon, require a new traffic management paradigm. Loon has worked extensively with other high-altitude industry players and regulators to support an international effort to define airspace management principles for the stratosphere, and we encourage others in the industry to continue to support this effort going forward.

Today's airspace management principles are built around the assumption that airspace users (i.e., pilots) do not have the ability (or technical means) to coordinate and deconflict themselves beyond the limited abilities provided by visual separation (under VFR rules). Centralized[2] Air Traffic Control services provided by Air Navigation Service Providers (ANSPs) ensure the separation of aircraft in congested airspaces and where visual separation is inappropriate. These services rely on an air traffic controller having situational awareness of the airspace via surveillance technologies (e.g., radar, transponder, and ADS-B) and communication with aircraft pilots (predominantly with radio voice communication).

While these services and technologies have proven to be extremely safe and robust through decades of experience, they face several challenges for managing large scale automated airspace operations:

- ATC services are built around the capabilities of traditional passenger aircraft, with the assumption of an onboard pilot. Pilotless, automated operations that behave in non-conventional ways (e.g., probabilistic trajectories) do not adapt well to existing ATC procedures and technologies.
- ATC services are expensive to provide due to the high infrastructure cost and their labor-intensive nature[3]. In most countries, Air Traffic Management service costs are paid for by fees paid by airspace users. The current cost recovery fee structure makes it challenging for stratospheric telecommunication services to operate profitably.

2 Centralized refers to a single entity providing separation services over an entire airspace (typically a country, or a group of countries).

3 For example, the FAA employs over 14,000 air traffic controllers.

New technologies that leverage the internet infrastructure, web protocols, and cloud computing, can be used to enable a new traffic management approach — decentralized collaborative separation ensured by the airspace users themselves. This approach has been first pioneered for low altitude operations in UAS Traffic Management (UTM). In a position paper[4] on Upper Class E Traffic Management (ETM), developed with industry's collaboration, the FAA[5] expanded this new framework to stratospheric operations stating:

"*ETM utilizes industry's ability to supply services under FAA's regulatory authority where ANSP separation services are not desired, appropriate, or available. It is largely a community-based, cooperative traffic management system, where the Operators are responsible for the coordination, execution, and management of operations, with rules of the road established by FAA.*"

UTM, however, needs to be adapted to the specifics of stratospheric operations. For the past few years, Loon led the stratospheric industry's concept development, and partnered with regulators, ANSPs, and research agencies worldwide, to define the needs of high-altitude traffic management and develop the framework. In October 2019, Loon presented to ICAO the industry's CONOPs for Collaborative Traffic Management in the Stratosphere (CTMS), which built upon NASA's early work. Loon engaged actively in the NASA-driven tabletops exercises, which are working towards a simulation demonstration. Loon also partnered with Airservices Australia to research CTMS protocols and participated on the advisory board of the European Concept for Higher Airspace Operations (ECHO)[6]. In May 2020, Loon led the creation of the industry Paper Adaptive Risk-Based Conflict Detection for Stratospheric Flight Operations presented at ATCA.

At a high level, the ETM construct (or CTMS–the industry's equivalent concept of operations) is like (and builds on) the UTM construct. It relies on stratospheric operators to collaboratively identify conflict and negotiate their resolution via automated exchanges of operational information (intended operational volumes, maneuvering performance, deconfliction thresholds, etc.). The notional entity of Platform Service Supplier (PSS) is introduced to designate the minimum set of services (of functions) necessary to operate within the ecosystem. PSS services may be provided by an operator itself (the operator is acting as its own PSS) or provided by a third party.

4 https://nari.arc.nasa.gov/sites/default/files/attachments/ETM_ConOps_V1.0.pdf
5 The FAA partnered with industry and NASA to develop the first version of the ETM paper.
6 https://www.eurocontrol.int/project/european-concept-higher-airspace-operation

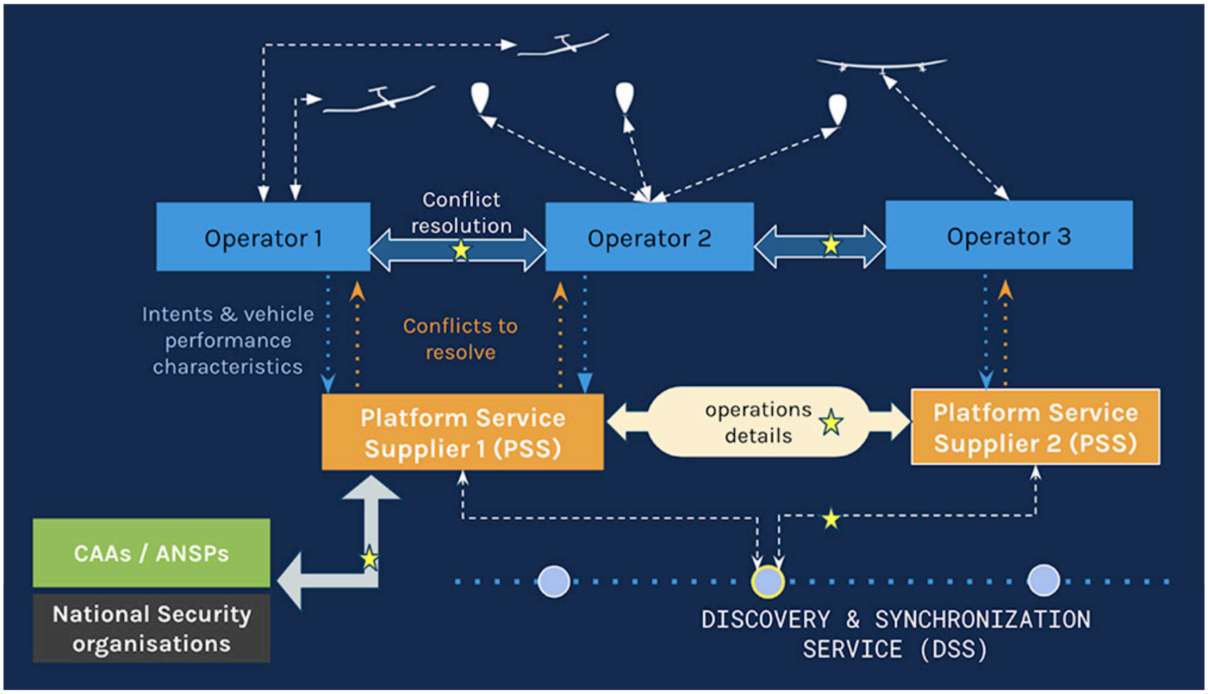


Figure 8-19 ***Overall CTMS framework presented at ICAO Drone Enable 2019.*** The framework leverages a federated collaborative resolution between airspace users. Each star indicates a standard protocol that needs to be developed. PSS functionality can be either performed by operators themselves or provided by a third party.

The CTMS framework can be decomposed into three categories of services, corresponding to the three main phases of deconfliction:

- ***Discovery and synchronization service (DSS).*** Like the UTM construct, the DSS protocol enables the various airspace users to discover one another and to subsequently exchange operation details to identify any potential conflicts. The DSS protocol provides a coarse mapping of which operator/PSS operates in a region that one would want to operate in, and how to contact these PSSs.

  The DSS protocol also provides the essential function of airspace versioning, ensuring that everyone has the latest version of the information.

- ***Conflict Identification Service (CIS)***. The CIS protocols enable PSSs/operators to exchange the specific operational details (operational intent, vehicle performance, and deconfliction thresholds) such that conflicts can be identified. The details are shared as needed following the discovery (DSS) process. The conflict identification protocols (and principles) in CTMS are significantly evolved from the UTM principles (and generalized) to accommodate the needs of the stratospheric operation.
- ***Conflict Resolution Service (CRS).*** The CRS protocols enable operators to negotiate a conflict resolution, once the conflict is identified. The outcome of the conflict resolution is a conflict-free state and associated changes of intents or commitments from one or both operators involved in the conflict.[7]

Two characteristics of stratospheric operations are driving the design of the CTMS protocols and the difference with UTM:

- ***Mission duration and characteristics.*** Stratospheric operations can last from hours to months (unlike UTM, which last for minutes). Many stratospheric operators replan actively during flight, and some like Loon have probabilistic trajectories with a growing uncertainty with longer horizons. This makes it impossible to reliably identify conflicts far in the future, and, like UTM, to perform deconfliction before take-off for the entire duration of a flight (potentially months long).
- ***Wide range of vehicle performances.*** The CTMS protocols need to accommodate operators and vehicles with extremely different performance and maneuvering characteristics (including new supersonic passenger transport, slow-moving fixed-wing HAPS, classic balloons with no altitude control, Loon like balloons with altitude control, and hybrid lighter-than-air vehicles combining light propulsion and altitude control). Not only the performance varies greatly between different vehicle types, but it may also vary in time for a specific vehicle (e.g., the performance characteristics may be a function of the state of charge of the battery). This means that operators (PSS entities) need to exchange potentially changing performance capabilities as part of the conflict identification protocol.

7 The burden of the conflict resolution maneuver may be shared between both operators. (e.g. Operator A increases altitude by 100 ft while Operator B decreases altitude by 100 ft).

As a result of the above, some core CTMS principles differ from UTM.

- ***Continuous forward-looking conflict identification.*** Unlike UTM, which deconflicts once for the entire duration of a flight, the CTMS conflict identification protocols use a continuous rolling time window approach (before take-off and while airborne). Operational intents are generated continuously for a specified forward-looking window of time.

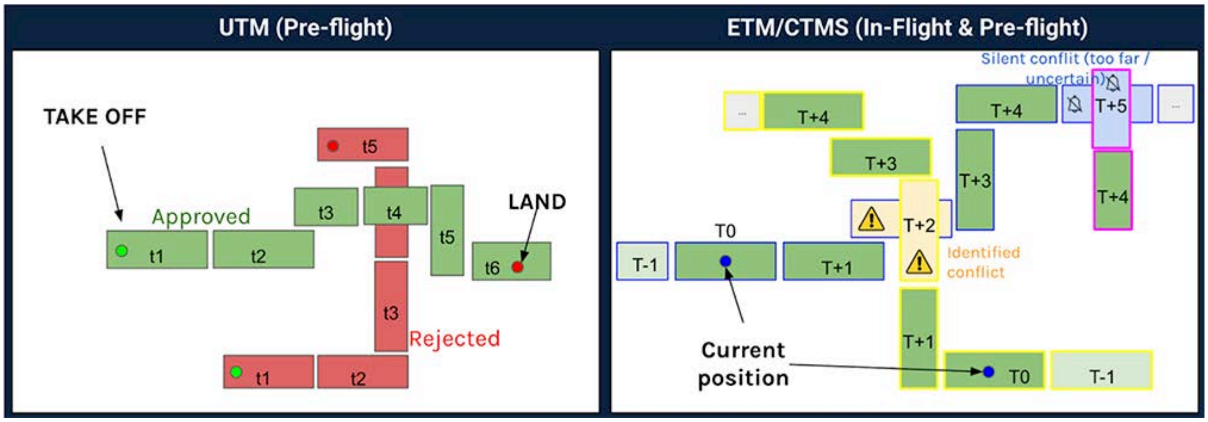


Figure 8-20 ***Conflict identification for UTM and ETM/CTMS.***
In UTM intents are deconflicted for the entire flight and 4D (space & time) overlaps are not permitted; the later submission is rejected while the first is approved. In CTMS, intents are shared continuously during flight, on rolling time windows. Conflicts are identified based on risk and imminence (how soon they will happen) and trigger a resolution negotiation.

- ***Collaboratively identified conflicts based on imminence and risk.*** In UTM, a four-dimensional (4D) intent overlap[8] constitutes a conflict. In CTMS, however, 4D intent overlaps do not necessarily generate a conflict, instead, a conflict is identified based on two criteria:
  - » ***Risk.*** A PSS continually estimates the likelihood of vehicles getting too close both laterally and vertically. For a conflict to be identified, the risk must exceed some threshold which depends on the likelihood of the excessive proximity and the severity of the associated undesirable event. To achieve target levels of safety, events with more severe consequences warrant attention at lower likelihoods.

8 Overlap of intended operations in both space and time..

» ***Imminence.*** Conflicts are identified depending on their imminence and the duration necessary to resolve them (time necessary to negotiate, communicate maneuver plans, execute maneuvers). A wait-and-see approach to distant events with low likelihood offers an opportunity for uncertainties to resolve favorably, avoiding the inefficiencies of premature flight adjustments.

While the exact protocol is still being researched, the currently discussed approach will require each operator involved in a potential conflict to exchange the timelines and likelihood thresholds necessary for avoiding the other party. These can be notionally represented as the following two-dimensional graph.

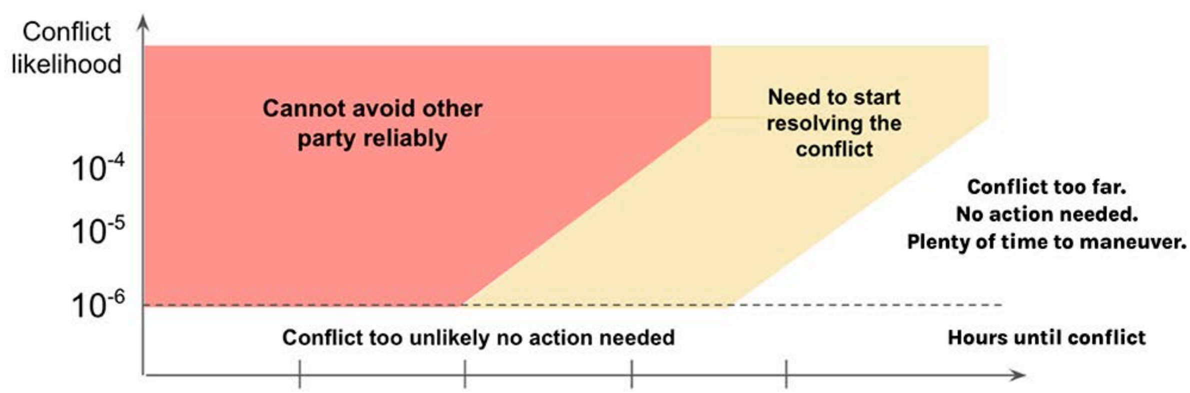


Figure 8-21 ***Operators communicate maneuverability/controllability and acceptable risk tolerance required to meet their safety case.***

The graphs from each involved operator can then be combined into a most conservative graph, used as a single source of truth by all parties to establish the deconfliction timeline. This ensures that everyone's safety case needs are met. The deconfliction negotiation process can start based on the timeline of the aircraft that is slowest to maneuver[9], even if the resolution does not ultimately require that maneuver.

9 This is necessary to ensure that either aircraft can safely maneuver out depending on the result of the resolution process.

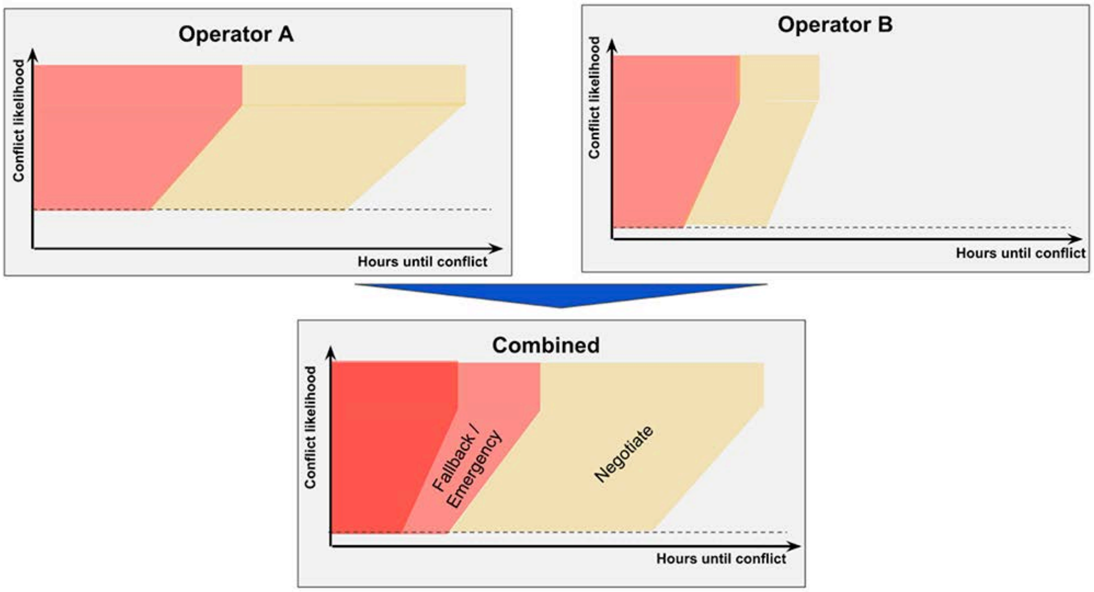


Figure 8-22 Combining conflict likelihood and timing graphs into a single most conservative graph.

The operators involved in the conflict also exchange 4-dimensional intents, which take the form of 3D volumes + time and have added probability contours[10] corresponding to multiple confidence levels. From these, the likelihood of excessive proximity can be derived and mapped on the graph above to identify a conflict (or not).

- ***Conflict resolution through negotiation, most likely powered by auction.*** When a conflict is identified, operators must resolve the conflict in a specific amount of time. The risk of an undesirable event (mid-air collision, wake turbulence impact, etc.) is reduced below an acceptable threshold.

  As intents are shared, and conflicts are identified on a rolling window, the conflict identification process must specify a time limit allotted for the collaborative conflict resolution. After the specified time limit is exceeded, the collaborative conflict resolution is deemed to have failed, and operators fall back on emergency rules of the road. Therefore, the conflict identification process must anticipate and budget for a conflict resolution timeline when deciding when to alert for conflicts.

10 This is specifically important for operators with probabilistic intents, such as balloons. For traditional aircraft, the probability contours can collapse to a single boundary.

The FAA requested that Industry players provide input to the design of the rules of the road. Fairness of access to the Airspace was a key driver in the industry's discussions. The working group recommended that a static priority rule (e.g., one based on vehicle maneuverability) is undesirable, unfair, and incentivizes inefficient airspace use. It was also suggested that a first-come, first-served approach is not appropriate for deconflicting on rolling time windows (no one is first). In addition, a first-come, first-served approach could incentivize greedy behaviors, which could be particularly challenging as some airspace users (such as balloons) have probabilistic intents, with the total volume of potential locations greatly increasing in the future. Instead, the industry working group recommended that conflict resolution leverage negotiation (accounting for everyone's constraints and priorities) and that the protocol should be designed to promote efficient airspace use. It is currently considered that the conflict resolution protocol would involve the following three elements:

» ***Free resolution protocol.*** Operators can leverage any bilateral agreement or protocols that they deem more efficient to resolve a conflict.

» ***Standard resolution protocol.*** A standard protocol that serves as a default in the absence of other agreed protocols and serves as a backup if the free resolution protocol does not yield an outcome. The standard protocol is likely to rely on an auction mechanism designed to promote fair airspace use and be designed to guarantee an outcome in a specified amount of time. The maximum time needed to execute this protocol will be documented in the standard.

» ***Emergency fallback rules.*** If a failure causes an operator to be unable to communicate, or the situation is such that a vehicle no longer can avoid the other one, emergency fallback rules will apply. It is worth noting that emergency fallback rules are designed to maximize safety, but they are not designed with fairness considerations. They are not a substitute for the standard resolution protocol and cannot be used by an operator instead of the latter to gain priority.

Each element can be mapped on the combined conflict identification notional graph.

Appendix A

Glossary

2T2R	Two transmit signals and two receive signals. Common LTE base station configuration.
3GPP	Third Generation Partnership Project. Governing body that creates standards for mobile connectivity such as LTE/UMTS/GSM. Basically, the org that defines the technical details behind the "G"s, like 3G, 4G, and now 5G (soon, 6G). See also http://www.3gpp.org/.
ACMB	Adaptive Coding, Modulation, and Bandwidth. Wireless signals can be designed to degrade gradually as the SNR (signal to noise ratio) decreases. This can be done in several ways: increasing the amount of error coding added to the signal (Coding), reducing the symbol rate (Bandwidth), and reducing the number of bits per modulated symbol (Modulation). ACMB is the ability to use all three of these techniques and choose the best combination as the SNR changes.
ACS	Altitude Control System (e.g., Franz, Thor) compressor used to change altitude by blowing air into or releasing air out of the air ballast chamber.
ADS-B	Automatic Dependent Surveillance-Broadcast
AE	Accountable Executive
AIA	Aerospace Industries Association
Airstream	A major component in Loon's LTE management system. Responsible for tweaking the more dynamic controls for each sector such as on/off, bandwidth, and transmit power, with the primary objective of a maximum number of users.
Alphadon	Prototype bus/payload for Jefferson/Hammerhead development.

AMDAR Aircraft Meteorological Data Relay

AnEn Analog Ensemble

Annex 2 Appendix 5
Rules of the Air - ICAO's Annex 2 to the Convention on International Civil Aviation. Appendix 5 covers the regulations applicable to unmanned free balloons.

ANSP Air Navigation Service Provider. A public or a private legal entity providing Air Navigation Services that manages air traffic on behalf of a company, region, or country.

Apex (Assembly)
The top part of the balloon through which gas was filled also contained the Flight Termination System, including the Apex circuit board.

Apex (printed circuit board assembly)
A PCBA on the balloon apex assembly that contained all the systems and components to control and fire the Flight Termination System (FTS).

ARTCC Air Route Traffic Control Center. Also known as Area Control Center (ACC).

ASAP map A type of Cartographer Map, used to navigate balloons to distant targets as soon as possible. See also Cartographer.

ASEAN Association of Southeast Asian Nations

ASECNA Agency for Aerial Navigation Safety in Africa and Madagascar

ATC Air Traffic Control

ATCA Air Traffic Control Association

Avionics All electronics involved in flight control, including computers, sensors, transponders, and satellite communications.

B2B Balloon-to-balloon backhaul communication.

B2C Balloon-to-customer

B2G Balloon-to-ground backhaul communication.

B2x The radio systems on the balloon payload that were used for backhaul (both B2B and B2G) and consisting of a gimbal+radio+antenna.

backhaul Communication path from a mobile base station to its core network. For LTE on Loon, this was the series of links from the LTE base station on the balloons (either Rickenbacker or Fodera) through the B2x mesh, through the ground station, then through fiber optics, back to ISP then through another fiber to the MNO's central office and the Loon Core NW.

ballonet A chamber (small balloon) inside the balloon used to separate the lift gas from the ballast gas (used to control altitude). Almost all other superpressure balloons fill the outer envelope with the lift gas (Helium) and pump air into/out of the inner ballonet to add or remove mass to descend and ascend. However, Loon reversed this to decrease helium leakage, filled the ballonet with helium, and used the outer compartment for ballast gas (air). French pronunciation is "bal-uh-ney".

balloon The balloon was the largest, most visible part of the overall Loon vehicle (or flight system) that also consisted of the envelope and ballonet, which held the lift gas and the air ballast gas, and the apex and bottom assemblies, that contained electronics and balloon-related mechanisms.

BBU Baseband Unit. The digital processing core logic for a mobile network base station.

BCI Basic Commerce and Industries. Partner with NCAR to provide NCAR-originated data to other entities, like Loon.

blister test One of the tests Loon performed to test film strength.

bottom assembly Opposite of the apex. An assembly at the bottom of the balloon that contained the tendon load rings, the ACS, and other electronics.

BRF Balloon Risk Factor

Brubeck	E-band B2x gimbal's controller and modem board.
bus	Part of the flight system that hung under the envelope and carried the service payload. Included power, avionics, Satcom, propulsion, and structure. Also, an electrical system term used to describe lines used in the Loon power system.
CAA	Civil Aviation Authority
CAN	Controller Area Network. A standard and very common communications protocol for industrial and automotive use. Loon used it for avionics communication within a balloon.
CANH	Controller area network high
CANL	Controller area network low
CANSO	Civil Air Navigation Services Organization
Capri	LTE baseband (modem) board for Rickenbacker. Replaced Wooten baseband from Fodera.
Cartographer	Loon software that created cartographer maps (both ASAP and Arrival time). Loon's primary mechanism used to navigate balloons to a distant target.
CASA	Civil Aviation Safety Authority
CASSOA	Civil Aviation Safety and Security Oversight Agency
CAT Map	Cartographer Arrival Time Map. A type of Cartographer map that was used to navigate a balloon to reach the specified target within a specified time window. Also, see Cartographer.
CDO	Convective Diagnostic Oceanic. NCAR/BCI field indicating the likelihood and strength of storm activity.
CDPI	Control Data Plane Interface (CDPI) protocol. Implemented as Airflow in Loon's Minkowski TS-SDN system.

CEP99 Circular Error Probability 99%. A method for monitoring historical performance of landing accuracy limiting outlier errors of more than 10km to only 1% of landings. The nominal goal is to keep the distribution shifted towards 5km without explicit distribution shape requirements.

CFD Computational Fluid Dynamics. Tools and algorithms for simulating and analyzing complex dynamics of moving fluids. Loon used CFD for the ACS turbo-pump and balloon design.

CIS Conflict identification service

CITEL The Inter-American Telecommunication Commission

CM Contract Manufacturer

CMM CNC image measuring machine

COCESNA Corporación Centroamericana de Servicios de Navegación Aérea

Comms Node Shared LAN switch, packet processor, and power distribution for all service payload systems. Aka “Gonzo.”

Conex Standard shipping containers. Often used for storage or as make-shift buildings.

CONOPS Concept of Operations. The overall description of how a large complex (typically aerospace) system operates and is operated, to achieve the objective for which it is being designed.

COTS Commercial-off-the-Shelf. Often just called OTS or off-the-shelf. Loon tried to use COTS parts where they worked. Generally, high-quality commercial parts were as good but cheaper than custom versions made to rigorous specifications, especially for high volumes.

CRM Crew Resource Management. Training, procedures, and philosophy for ensuring a well-functioning and responsive crew.

CRS Conflict resolution service

CTE Coefficient of thermal expansion. With such a wide range of temperatures, it's important to keep in mind how differently materials contract or expand. Matching the CTE is one way of minimizing this effect.

CTH Cloud Top Height

CTMS Collaborative Traffic Management in the Stratosphere. The industry's proposed concept for collaboratively managing and de-conflicting traffic at high altitudes.

CTU Caribbean Telecommunications Union

Daedalus Before it became Project Loon, Loon was internally called Project Daedalus.

Dark Crystal A Loon-developed automated hardware in the loop testing framework used to validate payload and GS hardware and software.

dB Decibel. A logarithmic unit that expresses a ratio, often of power or intensity.

dBm Decibel milliwatt. Measure of absolute power, in dB relative to 1mW. 0dBm = 1mW, -10dBm = 0.1mW, 2 0dBm = 100mW.

Despin A component used to decouple the natural rotation of the balloon from that of the bus to keep a specific azimuth of the bus pointing the solar panels at the sun throughout the day.

Desmond Interposer between the E-band radio (Wright) and the modem board (Brubeck) for the B2x subsystem.

DFMEA Design Failure Mode and Effect Analysis. A systematic group of activities used to determine how to recognize and evaluate potential systems, products, or process failures.

DFR Design for Reliability. The philosophy of incorporating reliability objectives into the early stages of the design rather than trying to address it later.

Downconnection

The long, slender aluminum structural tube that connected the balloon to the bus. Also, called "downconnection."

DPS Differential Pressure Sensor

DRA Diameter Routing Agent. A common element of mobile core networks.

DRBTR Design Review Based on Test Results

DSS Discovery and Synchronization Service. Protocol used in UTM and considered in CTMS that enables collaborative operators to discover each other and ensure freshness of every player's information state.

DWIR (or DIR) Downward Infrared. A measurement of the effective blackbody temperature below the balloon, which was an important part of Loon's thermal environment because of their use of photon harvesters to gather some of that heat at night. Also known as Upwelling IR. The aerospace industry sometimes calls this "Planet Shine."

E-band US FCC designation for the 71-76 GHz / 81-86 GHz band used by Loon's B2x and GS backhaul radios.

EASA European Union Aviation Safety Agency

ECHO European Concept for Higher Airspace Operation. The European CONOPs being developed by EUROCONTROL for traffic management at high altitudes. See also https://www.eurocontrol.int/project/european-concept-higher-airspace-operation.

ECMWF European Centre for Medium-Range Weather Forecasts. See also https://www.ecmwf.int/.

ECOWAS Economic Community of West African States

EFC Envelope Flight Computer. An additional flight controller, similar to PFC (see Primary Flight Controller) that was required for vehicles that supported the bus-separation descent system on v1.6. The PFC stayed with the bus and payload, but the EFC stayed with the balloon.

EIRP Effective Isotropic Radiated Power. A directional radio frequency power measurement where the EIRP is the hypothetical power that would have to be radiated by an isotropic antenna to give the same signal strength at a receiver.

EMF Electromotive force

EMI/EMC Electromagnetic Immunity/Electromagnetic Compatibility

eNodeB (eNB) Evolved Node B. An LTE "base station." The hardware at the user end of the mobile network that communicates wirelessly with mobile handsets (called User Equipment, or UEs).

envelope The outer layer (plastic film and tendons) of the balloon. This structure withstood the pressure differential with the ambient atmosphere. This is only the outer shell; the ballonet was not referred to as an envelope.

EPC Evolved Packet Core. The switching center of an LTE network. Loon's CoreNW contains two major components of the EPC: the MME and SGW. See also https://searchnetworking.techtarget.com/definition/Evolved-Packet-Core-EPC.

ESD Electrostatic Discharge

Estimator Loon software that ran either on the balloon or in the data center and used raw telemetry values from the balloon to estimate other more useful values. For instance, Loon estimated the altitude using several other sensor streams.

ETM Class E (high altitude) Traffic Management. See UTM.

EVM Error Vector Magnitude

FA Failure Analysis

FAA Federal Aviation Administration. See also http://www.faa.gov/.

FAS Future Access System

FCC Federal Communications Commission

FE Flight Engineer. Person in charge of supervising airborne operations.

FEA Failure Effect Analysis

FEMA	Federal Emergency Management Agency
FIR	Flight Information Region. See also https://www.icao.int/NACC/Pages/firs.aspx.
Fleet	The totality of Loon's airborne vehicles.
Flight system	Balloon + Bus + Payload.
Float	Float is a state that the balloon reaches after its initial ascent from the ground as it reaches equilibrium and stops ascending, at which point, "the balloon has reached float." Float also coincides with the end of the "pressurization phase" and leveling of superpressure.
Float altitude	Altitude at which balloon reaches equilibrium and stops ascending.
Flock	Group of flight vehicles associated with a single designated service area. A particular balloon may transition from one flock to another.
FMS	Fleet Management System. A collection of services, utilities, and user interfaces that enabled the fully automated operation of Loon's fleet of balloons. Loon's FMS (referred to in this document as FMS) was internally known as MC for Mission Control.
Fodera	LTE Gen1 system. Included Wooten (baseband + RFIC), Rex (receive RF + antenna), and Tex (transmit RF + antenna).
FPC	Flex Printed Circuit
Franz	One of the series of ACS that preceded Thor. See ACS.
FreeRTOS	Real-Time Operation System, which is open-sourced.
FSOC	Free Space Optical Communications. One of the balloon-to-balloon communications links studied before Loon deciding on the E-band wireless (RF) approach used by B2x and ground stations.
FTS	Flight Termination System. A system to intentionally bring down a flight by releasing lift gas to reduce buoyancy. Usually done with blades cutting through the top of the balloon.
FTSR	Flight Test Safety Review

Gadget	The gadget was a custom device (raspberry pi with Loon software) for administering, provisioning, and testing Loon payloads.
GB	Gigabyte (not gigabit). A common data amount unit.
Gbps	Gigabit per second (not gigabytes per second). A common data rate unit.
GDT	Gas discharge tubes
GEO	Geo-Stationary Orbit
GFS	Global Forecast System. NOAA's weather forecast and (re)analysis (historical) weather data service.
Gimbal	A pivoted support that allowed the rotation of an object in any direction in any plane. These were used for Loon's B2x and other mechanically directed antennas, including proposed future LTE systems for Loon.
GOES	Geostationary Operational Environmental Satellite
GOM	General Operations Manual
Gonzo	See CommsNode.
Gore	A single long, skinny diamond piece of balloon film that made up one vertical slice of a balloon. They were sealed together with their neighbors along the left and right edges.
GP	Gaussian Process
Gravity Waves	Transverse compression waves, or ripples, in the atmosphere. These result in regular oscillations of local atmospheric pressure, with periods from 5 minutes up to 20 hours. They cause an oscillation in wind direction, sometimes by 90° change in direction every wave period. Weather models are not able to predict these yet. They are commonly caused by convective storms and tall land masses disturbing the flow of wind. In the wake of the disturbing object, gravity waves can ripple downwind for hundreds of miles. Not to be confused with Gravitational Waves: disturbances in gravitation caused by big things colliding, such as black holes.

Ground truth The actual, real-world value. For Loon, when talking about winds, it was important to differentiate between forecasts, estimates, and assumptions versus the actual real winds. Loon rarely knew the ground truth for winds except in the vicinity of balloons.

GTEM Gigahertz transverse electromagnetic

HALO High Altitude, Low (Parachute) Opening. Descent technique that enabled rapid descent through the airspace, with a low altitude parachute opening, which landed the system at the desired low landing speed. This minimized airspace disruption and improved landing zone prediction accuracy.

HALT Highly Accelerated Life Test. A standard reliability testing term. A way to quickly discover operational limits and design weak points. Determined the effects of extremely high and low temperatures and vibration on the performance and functionality of the unit.

Hammerhead Next-generation Loon vehicles using a much larger and aerodynamic-shaped envelope to attain much higher availability, utilization, and revenue. The Hammerhead program pulled together the Jefferson program, which used an advanced, very strong multi-layer fabric to retain its aerodynamic shape, with the Raptor effort to create a similar, but cheaper and lighter but not quite as aerodynamic envelope using PE01 and tendons.

Hans Obsolete. ACS (Altitude control system) using impeller and two linear actuators to move impeller up and down to open/close valve. Replaced by Franz and then by Thor.

HAPS High-Altitude Platform Station, or High-Altitude Platform (plural)

Hard Terminate (HT) Flight termination mechanism that cut a large vent hole, which enabled a faster lift gas release and shorter descent time. See also Soft Terminate.

HASS Highly Accelerated Stress Screening

Hatchery Manufacturing/Assembly facilities at the Loon Launch Sites where flights were “hatched.”

HeLA Helium Launch Assist. A launch procedure that consisted of adding a small amount of helium into the air ballast chamber before launch (in addition to putting helium into the main helium chamber). This added some lift to accelerate the ascent and created separation between the envelope and the ballonet films, reducing the stress on those films. The helium was later removed from the air ballast chamber during ascent and float by releasing it through the ACS.

HIL Hardware in the loop

Hotel load The amount of power used to keep the balloon at minimum safe operation with no LTE or backhaul comms, no ACS or Seahorse to prioritize avionics and Satcom.

HRT High Resolution (or Rate) Telemetry transmitted via Loon's backhaul network (versus normal Loon telemetry, which is sent over SatCom).

Hydra (Loon) A group of battery cells in a pack; one of the sticks in a bus chassis is one Hydra. v1.4 had 11-12 Hydras and v1.6 has 12 Hydras.

I2C Board-level communication bus, often used for simple sensors.

Ibis Obsolete. Balloon generation "i", built using a "pancake" manufacturing technique.

ICAO International Civil Aviation Organization

IDP Inmarsat's IsatData Pro

IMU Inertial Measurement Unit. A device measuring orientation and acceleration.

Inmarsat Satellite communications service providing reliable low-data rate communications to phones and devices globally. One of two satellite comms services (SatCom) used by Loon for telemetry (from the balloon) and command data (to the balloon).

Iridium Satellite communications service providing reliable low-data rate communications to phones and devices globally. One of two satellite comms services (SatCom) used by Loon for telemetry (from the balloon) and command data (to the balloon).

Iron Sky	Container and mechanism that could release a commanded amount of powder ballast (iron powder). This replaced "single-shot" containers (box ballast).
Iron Snail	Before Loon was publicly known, it was known internally as "Iron Snail."
ITU	International Telecommunications Union
Kermit	The board that converts power from the solar panels using a Max Power-Point Tracker (MPPT) and charge controller.
Kestrel	An early Loon Zero-Pressure balloon, generally used for up and down same-day test flights.
Knuckle joint	The flexible joint that maintained the correct orientation of the bus even as the balloon envelope tilted.
KSI	Key Safety Indicator
LACAC	Latin American Civil Aviation Commission
LDPC	Low-Density Parity-check Code. A type of Forward Error Correction code used in telecommunications.
LEO	Low Earth Orbit
LFR	Lift and Fill Rig. The part of the launch system that held the Apex assembly of the balloon, lifted it, and filled the balloon.
LNR	Loon Network Router. See also ANR.
Loonix	The variant of ChromeOS (based on Linux) that ran on payload and ground station embedded computers.
LoS	Line of Sight
LTE	Long-Term Evolution. A standard for wireless communication of high-speed data for mobile phones and data terminals. An advanced form, now common, called LTE-A, is also known as 4G.

LZ Landing zone. The location where the balloon landed after the descent.

Major Tom The avionics firmware OS platform that ran on FreeRTOS and provided several key Loon avionics utilities and libraries. Used by 20 to 30 different MCUs across the balloon.

Mbps Megabits per second (not millibytes per second). A common data rate unit.

MBSE Model-based systems engineering

MC Mission Control. Loon's internal name for its Fleet Management System (FMS).

MCU Micro-Controller Unit. Microprocessor and key peripherals integrated into a single chip, typically containing some amount of data and code storage on the chip.

MD Meridional Direction. Thinking of the balloon as a globe, MD is in the north/south direction and along the tendon length. Also referred to as machine direction for the direction the film is extruded along the gore length. See also TD, which is transverse to this so similar to latitude lines on the globe.

mechanical fuse The component that protects the bus and payload from the extreme shock caused by certain worst-case balloon burst scenarios.

MEMS Microelectromechanical system

MEO Middle Earth Orbit. Satellites that occupy the large region between LEO and GEO.

Merlin An early Loon superpressure balloon design.

mil Unit of length, 1/1000th of 1 inch, typically used to measure the thickness of films, for Loon: balloon envelope film.

Minkowski	TS-SDN network operating system that controlled the topology and network routing of the Loon backhaul network, stretching from the Loon Core NW pods, through ground stations, and across B2B mesh. Ran in Google data centers.
MIP	Mixed Integer Programming
MJO	Madden-Julian Oscillation
MME	Mobility Management Entity. Manages mobility (among many other things) of a user (UE) across the network (part of LTE EPC). See also http://searchtelecom.techtarget.com/definition/Evolved-Packet-Core-EPC.
mmWave	Millimeter-wave. Radio signals in the ~20 GHz to 4000 GHz range. Loon sometimes used mmWave to refer specifically to their B2x radios that work in "E-band" (71 GHz to 86 GHz). The rest of the mobile industry uses "mmWave" to refer to 28 GHz to 39 GHz bands used for 5G.
MNE	Mobile Network Expansion. Loon's primary service offering: "cell towers in the sky."
MNO	Mobile Network Operator
MoC	Means of compliance
Moffett	Moffett Federal Airfield. Loon conducted a large variety of balloon and other tests in hangars and elsewhere in Moffett.
MPLS	Multiprotocol label switching
MPPT	Maximum Power-Point Tracking. A method of power conversion for solar panels.
MTTR	Mean Time to Recovery
NaaS	Network as a Service
nadir	The direction from an aircraft or satellite pointing directly down. The horizon then is 90° above nadir.
NAO	North Atlantic Oscillation

NASA National Aeronautics and Space Administration. See also http://www.nasa.gov/.

NCAR National Center for Atmospheric Research. See also https://ncar.ucar.edu/.

NFV Network Function Virtualization

NGSO Non-Geostationary Orbits. Non-stationary being key; these satellites move with reference to the Earth, so any terminal on the ground needs to track them or else needs to be sufficiently low gain to receive their signal from any upwards direction (but this means they are very low bitrate). Example: Iridium is a LEO satellite service, and all LEO and MEO (closer to earth than GEO) are NGSO.

Nighthawk Obsolete. Balloon generation “N” after “Ibis.”

NOAA National Oceanic and Atmospheric Administration. Source of some of Loon’s weather data.

NOTAM Notice to Airmen

occlusion 1) When air cannot escape the ballonet in flight. 2) Field-of-regard obstructions for B2x.

On-Call Responsibility of personnel (engineer or other) required to be reachable and available to assist (during and after business hours) when needed. On-call individuals have Service Level Agreements and need to respond within a specified amount of time.

On-Duty Responsibility of a Flight Engineer who is actively supervising the systems.

Operator Depending on the context, either an air services operator (e.g., an ATC) or a mobile network or communication services provider.

OQC Outgoing quality control

Oracle An FMS service predicting where the fleet will be located over the next few hours to days. Oracle performed an ensemble (assortment) of end-to-end fleet simulations using a randomized wind-noise model to predict the range of likely trajectories of the fleet (including network, power management, dispatching, and navigation control).

ORM Object relations mapping

ORT Ongoing Reliability Testing

Osprey Obsolete. Balloon design generation "O," similar to Nighthawk.

OTS Off-the-shelf

PA Power Amplifier

Part 101 14 CFR Part 101. FAA regulations applicable to Unmanned Free Balloons, the type of vehicle that Loon flew.

Pasture (balloon) Concept of operation used to safely maintain a flight vehicle airborne within a specified boundary with low/no population/air traffic. Used for long-duration testing of balloon reliability and/or UV degradation over time.

Path loss The power reduction (attenuation) of an electromagnetic wave as it propagates.

Payload Part of a vehicle's load, especially an aircraft's, from which revenue is derived. For Loon, the payload was an LTE and backhaul communication system.

PCBA Printed Circuit Board Assembly (vs. PCB, which is [officially] just the board itself, without components).

PCI Physical Cell Identifiers. An LTE parameter used to provide some amount of interference minimization between same-frequency mobile sectors. Assigning optimal PCI values to all of the static terrestrial sectors in an area is non-trivial but is more of an issue for Loon's moving sectors.

PE01 Loon's co-developed polyethylene envelope material used for all Loon balloons after 2017.

PFC Primary Flight Computer

PFD Power Flux Density. PFD threshold was used to determine if and how much signal can Loon "leak" outside of their intended serving area.

PFPE A lubricant used in the aerospace industry because of its chemical stability and stability across a wide temperature range.

Photon Harvest(er/ing)
Loon method of thermally regulating electronics by absorbing some of the infrared radiation from the earth's surface.

Plover Balloon design generation P after Osprey.

PLR Portable Launch Rig. Giant automated launching gantry crane.

Powerport Communications relay for comms node; controls power for LTE, Gonzo, and gateworks.

PR Puerto Rico. Usually in reference to Loon's launch site near Ceiba, Puerto Rico.

PSC Planning, Simulation, and Control team. A team of mostly software engineers (four of whom were physicists) who created almost all of Loon's data center software, mostly making up the fleet management systems. Included web user interfaces and data visualizations (Smallworld), steering controllers, long-range navigation mapping (Cartographer), fleet orchestration (dispatchers), wind data processing (WindGP, Wind Noise, Wind Mixer), Simulation engines and tools, estimators, automated exception detection, and much more.

PSS Platform Service Supplier. The notional entity providing a set of services required to operate as part of the CTMS Concept of Operations. A PSS may be part of an operator's capability.

PtP Point-to-Point (also called Pt2Pt and similar). Type of communications channel designed to communicate between only two devices at a time. For Loon, B2x and GS were PtP radios, while LTE was a P2MP (point to multipoint) system.

QBO Quasi-Biennial Oscillation. Periodic change in wind patterns. Quasi-periodic oscillation of the equatorial zonal wind between easterlies and westerlies in the tropical stratosphere with a mean period of 28 to 29 months. The alternating wind regimes develop at the top of the lower stratosphere and propagate downwards at about 1 km (0.6 mi) per month until they are dissipated at the tropical tropopause.

Quail The latest 'Q' letter generation of the balloon after Plover and used on v1.6 to support the additional mass of PDRA, Seahorse, a larger power system, and Dual ACS.

Rails, "The rails," RLS
X's building in Mountain View, California.

RAN Radio Access Network. A mobile network term that encompasses the base stations in an MNO's network, their backhaul, and the core network. Loon extended an existing MNO's RAN.

Raptor PE-01 film-based shaped envelopes designed for low drag and higher lateral speeds with Seahorse.

Raven Aerostar Large scientific ballooning company and long-time partner of Loons. See also http://ravenaerostar.com/.

Reverse ballonet
Using the ballonet for lift gas and using the rest of the balloon for the air ballast. Also called "Inverted ballonet." Loon's only balloon type from 2018 onward due to its significantly longer lifetime.

Rex Receive portion of Fodera LTE system.

RF Radio Frequency Communications

RL Reinforcement learning

Rickenbacker (also "Ricky")
LTE payload after Fodera that consisted of the Capri baseband and Soundblox RF boards.

RRH Remote Radio Head. In Loon, this most closely corresponded to the SoundBlox, which consisted of both a transceiver board and RF boards connected to the LTE antennas.

RSSI Received signal strength indicator

RTD Resistance Temperature Detector. A sensor whose resistance changes as its temperature changes.

Rx Short form of “receive” or “receiver”; used in communications engineering.

S2, or S2 cells Library of utilities, open-sourced by Google and used by Loon for working with earth coordinates. Loon used S2 cells to perform many spherical geometry calculations and manipulations in its navigation and fleet management systems.

SASP Separation and Airspace Safety Panel. An ICAO panel.

SBD Short Burst Data. One of Iridium’s data service offerings, which Loon used for their command/control link. Iridium offers other services with continuous connections.

Scutter Shuttle cutter. An alternative way to cut down balloons.

SDG Sustainable Development Goals

SDN Software-defined networking

Seahawk Fabric-based shaped envelope, initially designed for Loon’s Jefferson program but then re-focused to be the fabric envelope alternative for Hammerhead (with Raptor being the film-based alternative).

Seahorse A propeller fan, motor, and controls, to push the v1.6 (and later) vehicles horizontally (laterally) to assist the altitude-based (vertical) mechanism used to steer.

SEM Scanning electron microscope

SGW Serving Gateway. Routes data packets through the access network. See also http://searchtelecom.techtarget.com/definition/Evolved-Packet-Core-EPC.

SHT Super Hard Terminate. An FTS method developed but not used in production for generating very large vent openings for rapid descents.

SINR Signal to Interference + Noise Ratio

SLA Service Level Agreement

Sleepwalk Framework for incorporating reinforcement learning into Loon's steering (station-seeking) systems.

SLI Service Level Indicator

SLO Service Level Objective

Smallsky The staging version of Smallworld.

Smallworld Loon's front end (dashboard) for fleet management.

SME Subject matter expert

SMS Safety Management System

SMT Surface Mount Technology. Very small chips or components mounted directly on PCBs.

Snowth Backup battery for transponder only.

Soft Terminate (ST)
Flight termination mechanism that cuts a small hole in the balloon, causing it to vent more slowly than Hard Terminate. Backup systems to Hard Terminate. See also Hard Terminate.

SON Self-organizing network, as defined in the LTE spec. Several of the SON features were present in Loon's LTE Management System (Airstream).

Squib A self-contained pyrotechnic cutting device with very high reliability. Used for a variety of cut or release functions on the balloon. Electric current causes primary charge to heat and ignite. Commonly used as an actuator for parachuting applications.

SRE Site Reliability Engineering

StormTrooper The StormTrooper was a sensor board that collected data about storms and their effect on the flight vehicle. Included a lightning-detector, electric-field detector, corona-current detector, and moisture detectors.

Stratosphere The atmospheric region above the tropopause, typically from 15 - 50 km altitude, varies with weather/latitude/season. Loon balloons were nearly always in the stratosphere when at float.

SUA Special Use Airspace

Sub-assembly Sub-assemblies report to TLAs (top level assemblies).

Superpressure Balloon with higher than ambient pressure. At a given time, a balloon's superpressure is its pressure above ambient air pressure; an alternative name is delta-pressure.

Superpressure Balloon Oscillation
Regular oscillation in altitude around the neutral buoyancy pressure level. Occurs in superpressure balloons. Earlier Loon flights seemed to have an SBO period of between two and five minutes.

Super Soft Terminate (SST)
A third flight termination mechanism that cut a very small hole in the balloon. This triggered a very slow descent and very low landing speeds. See also Hard and Soft Terminate.

Supertemperature
Lift gas temperature - Ambient air temperature = Supertemperature. The balloon gas temperature relative to (above or below) the ambient air temperature. It can be negative at night (balloon gas is colder).

SWAP Size, Weight, and Power

Tack The point at which balloon tendon is attached to film to ensure film/tendon location relationship is good during and after the pressurization.

TD Transverse Direction. The direction of the balloon gore panel's width similar to East/West on a globe. Transverse to the direction of how the gore material is extruded (along the gore length).

TDR Time-domain reflectometry

TFOA Things falling off aircraft

Tendons Load-bearing high-strength cords strung from top to bottom of a superpressure balloon. Made of Dyneema or comparable materials, often bear more than 1,000 pounds of force each.

Tex Transmit (tx) portion of the Fodera LTE system.

TLS Target level of safety

TK Telkom Kenya

TROG A tester in the lab for testing scripts and hardware devices

TRON A Thermotron device for testing balloon payloads at low pressure and temperature. If a flight name starts with TRON, it was actually in a chamber in RLS.

Tropopause The atmospheric level between the troposphere (below) and the stratosphere (above), where temperature stops falling and starts increasing with increasing altitude. Height varies by season and latitude but is usually from 10-18 km.

Troposphere The atmospheric level from the ground up to the tropopause. The troposphere is defined as the region, starting at the ground, where the temperature decreases with increasing altitude.

TS-SDN Temporospatial software defined network

TTL Transistor-Transistor-Logic. Shorthand notation for a voltage range (not common now).

TVS Transient-voltage-suppression diode. Used to protect other circuitry from large spikes in voltage.

TWG Technical working group

TWR Time within range

Tx Transmit or transmitter. Used in the communications industry.

UAS Unmanned Aircraft Systems. See also https://www.faa.gov/uas/.

UE User Equipment. A device an end-user uses to communicate to a mobile network (most often, a mobile phone).

UHMPE (or UHMWPE)
Ultra-High Mechanical Polyethylene, or Ultra-High Molecular Weight Polyethylene. A very high tensile-strength material used for Loon's tendons and tethers. Used frequently across industries for very high-performance strength applications.

Upwelling Longwave Flux
Infrared heat energy radiating into space from the atmosphere + earth. Provided in watts/m^2, often casually expressed as a temperature in Loon. See DWIR.

UT Unplanned Termination (for balloons). User Terminal (typical for fixed wireless broadband).

UT30, UT60, UT24
An Unexpected Termination with less than 30 minutes/60 minutes/24 hours warning to Air Traffic Control ATC.

UTM UAS Traffic Management. See also https://www.faa.gov/uas/research_development/traffic_management/.

UTR Unplanned Termination Rate

UV Ultraviolet light

v1.0, v1.2, v1.3, v1.4, v1.6
Different generations of the bus/payload flew on different balloons that primarily represented evolutions in the service payload and power system.

VectorNav Maker of a pro-quality IMU used by Loon B2x to measure direction and attitude.

VLS Vertical Launch System. Similar to the PLR (see above), Loon's second-generation launch system that was completed late in 2020 to accommodate much larger balloons and enable vertical launching of very tall balloons.

VNF Virtual Network Function. Loon used several VNFs in the Loon Core NW, including a virtualized MME, SGW and DRA, and Loon's own LNR (see above).

VoLTE Voice over LTE. Though LTE was the next-generation mobile phone standard after 3G, it did not support voice calls initially, relying instead on the phones switching to 2G or 3G to make or receive calls. VoLTE was added later to avoid the need for that fallback to the older 2G or 3G networks. This was important for Loon because many of their service areas had no 2G or 3G coverage.

VSFEI Very Small Fleet Experiments Infrastructure. Full dynamic fleet simulator. VSFEI replaced a more difficult way of accessing such sims through a combination of the colab notebook and other back-end tools.

VSP Very Simple Planner

WMC Winnemucca, Nevada. Loon launched many balloons from there, thanks to a great collaboration with the community over many years.

Wooten Baseband and transceiver PCBA for Loon's Gen1 Fodera LTE base station.

Wright The E-band B2x radio board PCBA. It performs upconversion to E-band (71-86 GHz) from baseband for transmitting and downconversion from E-band for receiving.

Yin-yang Obsolete. A ballonet design starting in July 2016 that had the ballonet on its side compared with the Ibis balloon. Replaced by reverse ballonet architecture in 2018.

Zero-pressure (noun)

Balloon designed to have the same lift-gas pressure as the air pressure outside. Will descend after sunset due to cooling if ballast mass is not dropped. Loon only flew a few in the very early days.

Zero-pressure (verb)

When a superpressure balloon (such as any Loon balloon) loses positive pressure due to temperature dropping. Followed by descent unless ballast is dropped. Loon tried to stay away from this. Also called “ZP” or thermal runaway.

Appendix B

Loon Accomplishments

<table>
<tr><th colspan="2">Accomplishment</th><th>Total</th><th>Notes</th></tr>
<tr><td colspan="2">Number of flights</td><td>2127 flights</td><td></td></tr>
<tr><td colspan="2">Total flight time</td><td>1,915,696 hours
79,821 days
218.5 years</td><td>For all flights
Hours to days
Days to years</td></tr>
<tr><td colspan="2">Total flight distance</td><td>>70,000,000 km</td><td>For all flights</td></tr>
<tr><td colspan="2">Longest flight by distance</td><td>217,260 km</td><td></td></tr>
<tr><td colspan="2">Longest period aloft</td><td>336 days</td><td></td></tr>
<tr><td rowspan="6">Average flight duration</td><td>2016</td><td>50 days</td><td>Landed in 2016</td></tr>
<tr><td>2017</td><td>68 days</td><td>Landed in 2017</td></tr>
<tr><td>2018</td><td>103 days</td><td>Landed in 2018</td></tr>
<tr><td>2019</td><td>114 days</td><td>Landed in 2019</td></tr>
<tr><td>2020</td><td>142 days</td><td>Landed in 2020</td></tr>
<tr><td>2021</td><td>161 days</td><td>Landed in 2021</td></tr>
</table>

Appendix C

Loon Specifications

			Hammerhead	
	v1.4	***v1.6***	Gen 1	Gen 2
Balloon				
Balloon version	***Plover***	***Quail***	***Raptor***	***Seahawk***
Material	PE film	PE film	PE Film	Fabric
Minimum operating altitude	16 km	16 km	15.25 km	15.25 km
Maximum operating altitude	19.5 km	20.5 km	20.7 km	18.5 km
Balloon diameter	8.52 m	9.85 m	29.5 m	22.8 m
Balloon length	–	–	60 m	57 m
Volume of envelope	1800 m^3	2,713 m^3	26,460 m^3	15,050 m^3
Max lifting capability[1]	226.8 kg	261.2 kg	1875 kg	1340 kg
Propulsion				
Nominal Lateral Power	–	70 W	2300 W	1800 W
Average maximum sustained speed[2]	–	1 m/s	5.25 m/s	7.8 m/s
Number of propellers	–	1	2	1
Diameter of propeller	–	2	2.5	3
Other balloon attributes				
Maximum ballast	25 kg	25 kg	not final	not final

1 Dry mass for Hammerhead systems

2 Assuming 30% of total energy harvest for prop; estimated

			Hammerhead	
	v1.4	v1.6	Gen 1	Gen 2
Power system				
Battery S/P config (per pack)	12s2p	12s6p	not final	not final
Battery pack type	Hydra	Triplehydra	not final	not final
Number of packs	10	5	not final	not final
Total Energy Generation	9267 Wh	9730 Wh	129,000 Wh	66,500 Wh
Total Storage Capacity	2764 Wh	4146 Wh	73,800 Wh	43,300 Wh
Solar panels				
Total Panels Area	4.6 m^2	4.6 m^2	68.25 m^2	36 m^2
Daily harvest at equator equinox	9267 Wh	9730 Wh	129,000 Wh	66,500 Wh
Payload				
B2X • Type	RF wireless	RF wireless	RF wireless	FSOC (laser)
• Frequency band	E-band	E-band	E-band	—
• Data rate per gimbal	1 Gbps	1 Gbps	2.5 Gbps	10 Gbps
LTE • Fixed sectors	4	4	4	7
• Gimbals	0	0	3	3
• Sectors per gimbal	—	—	7	7
• Total sectors	4	4	25	28

Appendix D

List of Figures