CloudSat

Centralized-Adaptable Ground Station Architecture utilizing an IoT Design Methodology

Mr. Brian Zufelt
COSMIAC
University of New Mexico
Albuquerque, United States
Brian.Zufelt@cosmiac.org

Dr. James Aarested
COSMIAC
University of New Mexico
Albuquerque, United States
Jim.Aarested@cosmiac.org

***Abstract*— Over the last decade, the cost of space access has dramatically decreased with the creation of the CubeSat standard. The CubeSat standard defines the structural requirements for an on-orbit deployer and satellite to be placed into orbit. The average cost of creating a space mission with the CubeSat standard can range from $200 thousand to $3 million. This lower cost has allowed many Universities, and small businesses to create their own space programs. However, a significant portion of the investment for any new space asset is the development of the ground system to communicate with the satellite. These costs can be further reduced by removing the need to develop a custom communication solution. CloudSat is a scalable IoT architecture designed to grow and adapt to the needs of low-cost small satellite missions by providing a central market of networked ground stations, tools, and resources for communicating with on-orbit assets. In addition, CloudSat provides a marketplace where groups can lease time on their space assets to groups that lack the resources to invest in the development of a new space asset. The goal of CloudSat is to provide a means of gaining access to space technology to anyone who can benefit.**

Keywords—CubeSat; Ground Station; IoT; Satellite

I. Introduction

The ability to maintain constant communication with a satellite has been an increasing need for the CubeSat community. The current CubeSat designs are able to gather and process large amounts of information without a way to get this information to the ground. Since CubeSats are small (4in cubed), compared to other satellites, they are able to fly very close to earth in an LEO orbit (Low Earth Orbit). LEO orbits are a relatively safe environment for terrestrial electronics given a reduced amount of radiation that the satellite will encounter. For this reason, CubeSats can benefit from the advancements made in the commercial electronics industry. This keeps the cost of CubeSat development low, while still taking advantage of the latest technology. Larger satellites have to adopt new technology to survive the space environment before new components can be used. This process can take years and cost millions of dollars.

While the CubeSat standard has reduced the cost of developing and deploying a space asset, the cost of the ground station, to communicate with the asset, remains a significant cost. In addition, the ground stations that have been developed are significantly underutilized. Some for only twenty minutes a day. By reducing or eliminating the cost of a ground station for a new mission, more communities could enter the space market. CloudSat provides a way to reduce the cost for a single mission while opening a new market to sell access to space. This paper outlines the CloudSat network and how it can increase utilization of our current ground station infrastructure while reducing costs.

A. Problem with current Ground Station Designs

The common CubeSat design is limited to a single UHF/VHF/S-Band ground station with a transfer rate from 9600bps to 3Mbps, and a couple passes for ~7 min a day.[1] In addition, more substantial data rates require more costly infrastructure that may not fit into the mission budget. For example, to achieve 3Mbps on UHF requires a 20-meter dish to close a 1-watt link. Thus, this system does not allow for large amounts of data to be transferred, or for critical health and status messages to be delivered in a timely manner. Compression of data will increase throughput but does not allow for time critical messages to be delivered to the operator with enough time to mitigate failure. The typical ground station design not only limits the type of missions CubeSats are used for, but also reduces the reliably of the missions flown. The common ground station configuration is shown in Figure 1. Each ground station consists of two main elements, the *Ground Station Control* (GSC) terminal, and the *SatCOMM* (SCT) terminal. The GSC is responsible for making sure the system is tracking the satellite correctly and adjusting the radio to compensate for Doppler effects. The SCT is were messages are received and processed. The SCT controls the Satellite, while the GSC controls the ground station.

This current situation poses three main problems. First, the existing infrastructure of ground stations are under-utilized. Second, the reduced footprint on the ground limits the contact time with the space asset, and thus, reduces the data budget for the mission. Finally, the current single ground station infrastructure significantly increases the cost of new missions and can limit the communication options available to some teams. By connecting the current ground assets to a central control system, the ground footprint will become larger which will allow for longer contact times, more data to be downloaded, and open up new communication technologies to groups that didn't have access due to budget constraints.

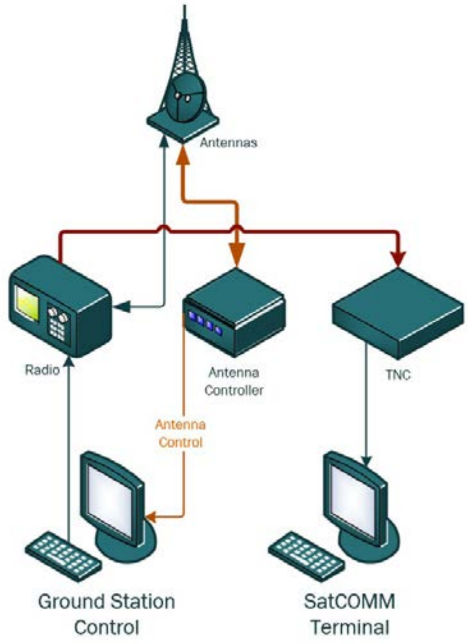


Figure 1:Average Ground Station Configuration

B. *Early Centralized Ground Station Management designs*

The CubeSat community is growing every day. Over a hundred satellites were launched globally in 2016, and even more in 2017. All of these missions require a ground station. The majority of US CubeSats used a dedicated ground station.[2] Which means these ground stations were only used for about 15min a DAY! If there was a second option to the current ground station design, that could provide global coverage, no maintenance requirements, and increased reliability, the US CubeSat community would be more than glad to switch their current condition of operations (CONOPS) for new missions. In addition, there are many ground stations currently not being used at a hundred percent capacity. Repurposing these ground stations for the CloudSat network would increase the power and reach of the CloudSat system while achieving more utilization of the ground system.

The idea of a central ground station control for Nano-Satellites is not new. One of the original designs was from a Stanford student who studied under Prof. Twiggs (co-inventor of the CubeSat standard). His concept was to create a central control center that would task ground stations remotely. However, this tasking was only for the GSC. This means the system would only control the ground station for receiving packets and place all data collected on a local computer. It was the responsibility of the local station management to send the data to the satellite team. As you can imagine, this created many problems with specific missions that required time-sensitive data to be passed and exposes sensitive data to compromised systems.

An expansion of the Stanford design was completed in 2010, by the European Space Agency (ESA). This concept was known as the Global Educational Network for Satellite Operations (GENSO).[3] Figure 2 shows the map of the GENSO ground station network.

COSMIAC at The University of New Mexico was one of

Figure 2: GENSO Ground Network

the first ground stations to support GENSO missions in the United States. The GENSO design was an improvement on the Stanford design. GENSO tasked ground stations to control tracking, and it was able to receive data and place it in a central repository for satellite operators to receive. However, GENSO suffered from a lack of ESA support and only allowed for one set of equipment. This hindered its acceptance since many ground stations want the freedom to choose their own equipment. GENSO was just able to track a single satellite within a region. Meaning if there are two assets over the same footprint, GENSO would task all the stations within that footprint to follow a single asset.

CloudSat was designed to solve the limitations of previous systems. Limiting the satellite developer will prevent the CubeSat community from embracing the benefits of a networked ground architecture.

II. CLOUDSAT DESIGN

The key to the CloudSat concept is to allow for flexibility in CubeSat and ground station design. CloudSat should be seen as a type of Facebook for spacecraft. Figure 3 shows a basic block diagram of how CloudSat can integrate into a single ground station. The TNC (Satellite Modem) is connected to a SatCOMM relay that connects to the CloudSat system. This relay box will be used for posting data to the database and receiving two-line elements (TLE) for satellite tracking.

By creating multiple-opensource paths for data to be transferred to and from the CloudSat server, satellite operators are free to choose what equipment and components they want to use in their ground station design.

Once the connection to CloudSat has been made, the real benefit to this approach becomes apparent. Payload developers (PD), which do not care about the satellite as a whole, are able to work with their experiment directly without going through the satellite operator. The Satellite Operator (SO), who is in charge of keeping the satellite functioning, acts as an administrator of the satellite. The SO will be able to turn on and off payloads and not have to worry about their operation, unless notified by the PD. The Scientist will have immediate access to the data from the payload. So, they will be able to move forward with their work without going through the PD. Finally, another Satellite Developer (SD) can learn from the performance issues of earlier missions. Other SDs will be able to see what has worked in the past and develop more reliable satellites in the future. The real power of CloudSat is that it allows all elements of a satellite team to work in parallel and not through a chain.

Another benefit to the CloudSat concept is the ability to manage multiple missions at once. For example, a cluster of satellites can measure a solar event. These satellites will be scattered in various orbits. Since all the data will end up at a central location, the scientists will be able to work with these satellites as if they came from a single satellite.

CloudSat will not only be more flexible in the choice's developers have to make in their design, but it will also actually increase their options. If a node on the CloudSat network can work in X-band, S-band, UHF, VHF, Laser COMM, Iridium, Global Star, TDRS, etc.; the SD can choose a communication system that works with any of these ground stations and not have to worry about purchasing the equipment to communicate on these channels. The SD will just communicate through CloudSat. This means that as more ground stations are brought within the CloudSat umbrella, the more benefits CloudSat will provide the community.

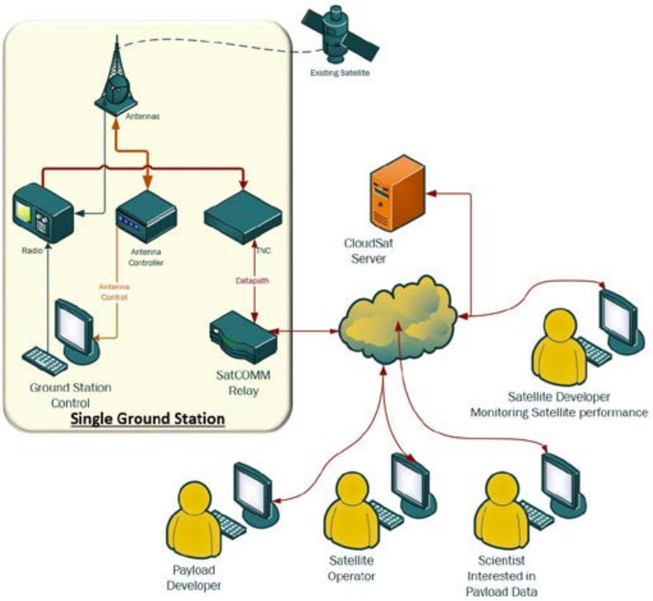


Figure 3: CloudSat Connection Overview

The ability to use another group's ground station opens up opportunities for communities to gain access to space. In addition, satellite owners can task their on-orbit assets to realize the goals of new missions. Provided they have the necessary sensors and payloads. The ability to use existing assets to achieve the goals of more modern mission concepts could significantly reduce costs and further open up space to communities that cannot afford a launch or ground station.

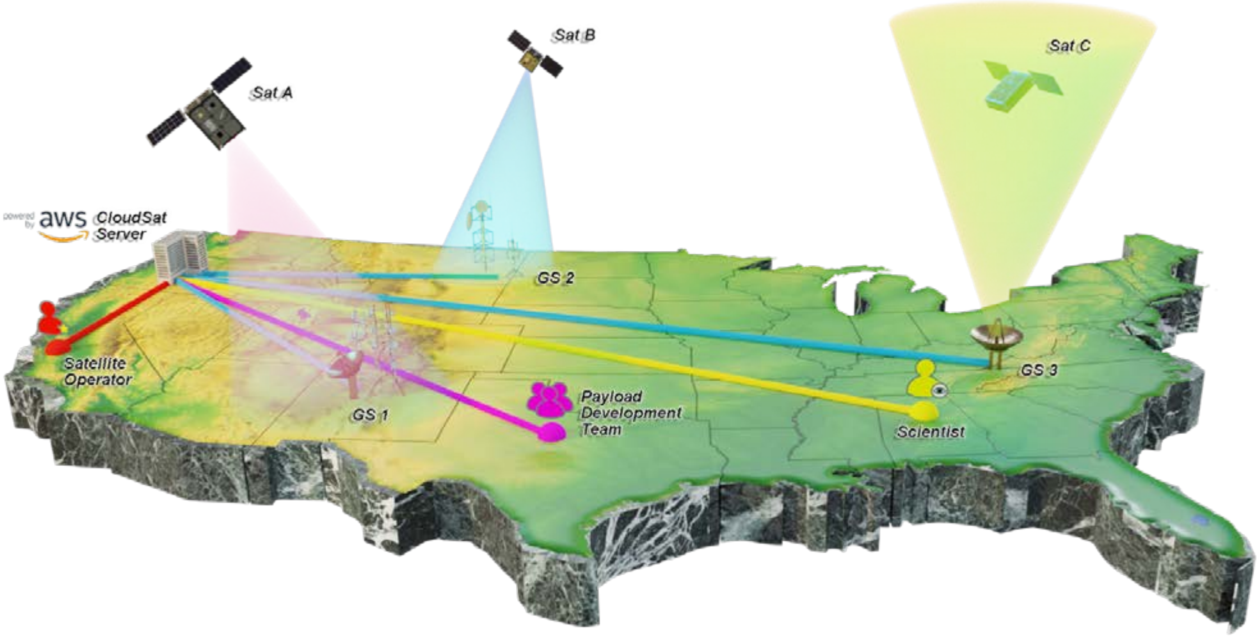


Figure 4: CloudSat CONOPS

A. Conditions of Operations (CONOPS)

CloudSat is a web application intended to aid in the development and management of space assets, by providing a common ground architecture interface. The central web application is hosted on the Amazon Web Services (AWS) platform. In basic terms, CloudSat is designed to be a simple pipe for the space asset to deliver data packets to a “bucket” (MySQL) housed on AWS. Keeping to this simple design focus forces CloudSat to maintain a simple interface for the end user.

CloudSat is designed to follow the space asset from design, through launch, to on-orbit operations. Initially, the SD will set up an account and define the roles of additional users (payload developers, scientists, etc). The SO will be the administrator for the account. Since the idea of CloudSat is to move the appropriate data from a ground station to the database associated with a given account, CloudSat can be used to test different packet formats during the development stage. Using a “dummy” ground station application to pose as a legitimate ground station, the satellite developer can inject FlatSat data into the network during day-in-the-life testing, or try different packet formats to confirm data budgets.

After launch, the space assets Two-Line-Elements (TLEs) are provided by the SO. The TLEs are used to track the asset, and when mixed with other TLEs from other accounts CloudSat will create a tasking list to be sent to the ground station. The tasking list is used by the ground station to determine when an asset is going to be within range and what commands it should send and how much data it is expected to receive. The tasking list also includes the ID of the account for the requested transactions. The ID allows all packets received to be forwarded to the associated account. Once the account receives the packet it is entered in the database with flags to indicate access level for the users within a given account. CloudSat associates accounts with space assets and not single missions or satellites because it is not limited to a single mission or satellite. Imagine the scenario shown in figure 4. The mission could involve a cluster of satellites (Sat A,B,C). Sat A,C could use the same radios. However Sat B is an older satellite and uses an older radio. In addition, the download requirements are greater than what is possible on a single ground station. Finally Sat A,B are over the same ground station footprint at the same time. Which means a single ground station would struggle to communicate with both assets during one pass. Within the CloudSat architecture however, the three ground stations are able to meet requirements of the mission by creating three separate tasking lists that are passed to GS1,2,3. Sat A,B can download their date during pass on GS1,2 while Sat C is receiving commands from GS3.

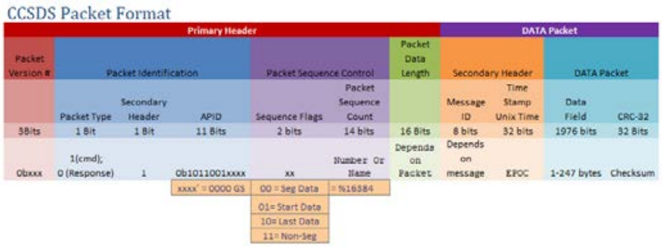
B. Protocols and Data Distribution

Different protocols are required to realize the ground system infrastructure CloudSat is trying to achieve. It is easier to split the system into two communication channels. First is the communication between the ground station and the on-orbit assets. Second is the communication between the ground station and the CloudSat central data center.

1) Satellite to Ground Station

CloudSat connects to the ground stations over a TCP/IP internet connection, so communication between the ground station and the satellite will always be limiting “bottleneck” of the network. Also one of the CloudSat goals is to make communication as flexible as possible. Unfortunately, this does not allow for every protocol to work on the network. Since a ground station to satellite protocol had to be chosen, the CloudSat team choose the Consultative Committee for Space Data Systems (CCSDS 133.0-B-1) standard.[4] The CCSDS standard has been widely used for many missions and high-altitude balloon flights. Given its wide acceptance, it was the decision that CCSDS would require little, to no modifications to current spacecraft designs.

Figure 5 shows the space packet protocol as defined in CCSDS 133.0-B-1.


CCSDS Packet Format

Primary Header							DATA Packet			
Packet Version #	Packet Identification			Packet Sequence Control		Packet Data Length	Secondary Header		DATA Packet	
	Packet Type	Secondary Header	APID	Sequence Flags	Packet Sequence Count		Message ID	Time Stamp Unix Time	Data Field	CRC-32
3Bits	1 Bit	1 Bit	11 Bits	2 bits	14 bits	16 Bits	8 bits	32 bits	1976 bits	32 Bits
Obxxx	1(cmd); 0 (Response)	1	0b1011001xxxx	xx	Number Or Name	Depends on Packet	Depends on message	EPOC	1-247 bytes	Checksum
			xxxx' = 0000 GS	00 = Seg Data	= %16384					
				01= Start Data						
				10= Last Data						
				11= Non-Seg						

Figure 5: Satellite to Ground Station Format

The CCSDS packet contains two sections a *Primary Header*, and *Data Packet.* The primary header contains all the information about the destination of the packet and the size of the payload. The secondary header, found in the data packet, contains the specifics of the data to be transferred.

While this paper does not explain the full CCSDS standard, the primary header contains two key sections that are worth noting for the services they provide. First, the Application Process Identification number (APID) identifies the recipient of the packet. The Command and Data Handler (C&DH) of the spacecraft is the central processing element of every CubeSat. When the C&DH receives a CCSDS packet, it looks at the APID and forwards the packet to the intended recipient. The APID could be another payload within the spacecraft or the ground. To ensure the ground station is not confused with packets traveling through it, the APID for ground station must always be 0b10110010000. The APID gives control to the end user to send packets to not only the spacecraft, but also a specific payload within the spacecraft.

The second service to note is from the *Sequence Flags* found in the primary header. The sequence flags allow data to be split into multiple packets for transmission. When a packet is to be split the first packet is sent with a flag of 0b01. All subsequent packets will have a flag of 0b00, until the last packet contains 0b10. The sequence counter will ensure that no packet is lost in the transmission.

2) Ground Station to CloudSat Server

Given the small packet size of the CCSDS standard, and that the spacecraft's function is similar to a sensor found within a clustered sensor network, an Internet-of-Things (IoT) approach was chosen for the network protocol between the ground stations and the CloudSat server. A common IoT protocol was used. Message Queuing Telemetry Transport (MQTT) is an ISO standard (ISO/IEC PRF 20922)[5] used for many IoT applications. MQTT is publish-subscribe-based messaging protocol. The MQTT clients publish or subscribe to data channels from a MQTT broker. The broker is tasked with receiving packets and placing them in the appropriate data channel, so they are available to the clients. Figure 6 outlines the MQTT packet format.

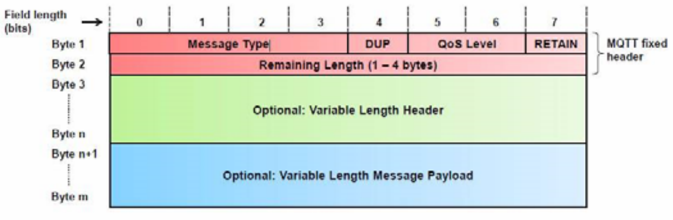


Figure 6: MQTT Packet format

As with the CCSDS standard, this paper is not intended to cover the full MQTT protocol. However, there are some key elements to outline. First the *Message Type* contains the control tag for the message. The common tags are connect, disconnect, publish, subscribe. Second, the *quality of service level* defines the importance of the packet. CloudSat uses the QoS level to prioritize missions so that ground stations can switch between different space assets for high priority missions. Finally, the last section of the packet contains the full CCSDS packet to be transmitted or received from the space asset. The CCSDS packet is encapsulated in the MQTT packet format.

The CloudSat server performs all broker services defined by the MQTT protocol. The ground stations and the CloudSat end user web interface function as the clients. The ground stations must subscribe to their individual tasking account to receive tasking commands, TLE tracking information, and packets to be transmitted. For packets to be transmitted, the ground station is to transmit the CCSDS packet as received with no preprocessing. Ground stations must also publish all packets received to their own receive channel so that CloudSat knows which ground station received a given packet for processing.

The CloudSat end user web interface subscribes to the channels defined by the user's accounts. When a user sets up an account they define the channels for which packets will be received and where transmitted packets will be transmitted.

C. Ground Station Control

The CloudSat core is responsible for tasking the ground stations, and placing received data to the database for access to the end users. The ground station receives the tasks through a secure link. While a software suite could be created to manage the link between the CloudSat server and the ground station, a hardware approach was chosen to allow for easy integration into most commercial ground stations. Some larger ground stations, like the 20-meter dish at Wallops have strict security requirements and do not allow unauthorized software to run on their servers. However, the Wallops group allow for a serial connection to user hardware. For situations like this the CloudSat team created the SatCOMM Relay (SCR). The SCR function is to collect packets received by the radio, send packets to the radio for transmission, and relay tasking lists to the tracking software. Version 1 of the SCR can be seen in figure 7. This system houses an ARM Linux computer with a custom daughter card to extend the interface to include TTL UART, RS232, RS 422, I2C, USB, and SPI. In addition, the TFT display relays important performance and connection status to the ground station operator. The SCR is intended to be small and contain as many interfaces as possible to ensure compatibility with as many GS systems as possible.

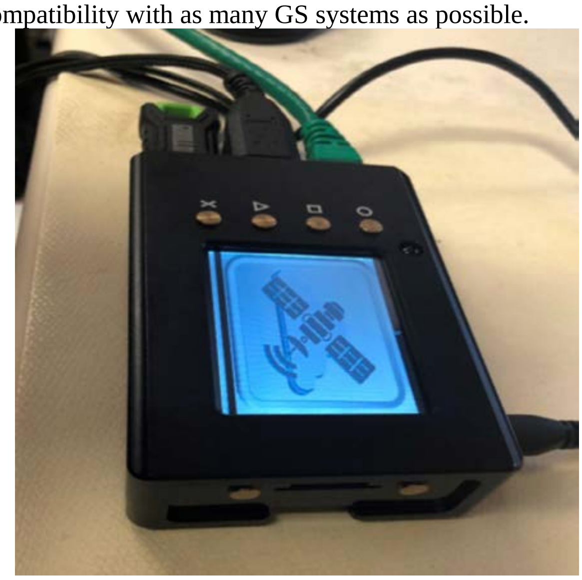


Figure 7: SatCOMM Relay Controller (SCR) Version 1

Another connection the SCR must make is to the ground station control terminal. This connection is developer's choice and is only used to pull a list of TLEs to task the ground station to track a particular satellite.

D. Security

Security is a critical part of the CloudSat system. However before discussing the security CloudSat provides, it is important to mention the areas not addressed by the CloudSat system. First, the choice to encrypt telemetry and control from within the payload of the CCSDS packet is the choice of the SD. CloudSat only secures the communication channel from the end user to the ground station. This limitation allows the SD the flexibility to secure their space asset to meet their mission requirements. The ground station will transmit the CCSDS packet as received with no preprocessing or packet authentication. CloudSat provides security services between the ground stations and the CloudSat server, in addition to the end user and their account.

1) *Ground Station Link Security*

When the ground station account is created it will receive security certificates from the CloudSat certificate authority. These certificates are RSA 4096-bit key pair. The ground station will receive its public and private certificates with the CloudSat public certificate. The ground station will use the certificated to encrypt and sign all MQTT packets sent from the ground station to the CloudSat server. The server will encrypt and sign all packets it sends to the ground station with the ground station public certificate and its own private certificate. Signing each packet allows for each end of the link to authenticate the source.

2) *User Security*

The user space is secure using HTTPS SSL certificates to authenticate the web interface and create an end-to-end link through the user's browser. If the command and telemetry frames are encrypted by the spacecraft and end user, they will remain encrypted until the user downloads the data through the web interface.

III. Conclusions and Future Work

There are many benefits to the CloudSat network that are still to be explored. One of the most disruptive features is in the ability to create a "virtual" satellite mission from existing space assets. Virtual satellite missions can be created by simply by creating a tasking list for different ground stations. These ground stations will task different satellites to perform operations that would have been performed by a "real" version of the virtual satellite mission. The concept of a virtual mission allows for communities that do not have the means to develop a satellite, access to space. Creating virtual satellites or leveraging current ground stations, new markets can be created where groups can share underutilized resources with less fortunate communities.

Finally it is the goal of the CloudSat project to create a community of satellite teams to share resources and reduce the operating costs of future space programs.

References

[1] T. Choi, T. Stevenson, G. Lightsey. (2017). "Reference Ground Station Design for University Satellite" AIAA SciTech Forum 55th AIAA Aerospace Sciences Meeting

[2] M. Fischer and A. L. Scholtz, "Design of a Multi-mission Satellite Ground Station for Education and Research," *2010 Second International Conference on Advances in Satellite and Space Communications*, Athens, 2010, pp. 58-63. doi: 10.1109/SPACOMM.2010.13

[3] G. Shirville, B. Klofas, (2007) "GENSO: A Global Ground Station Network" AMSAT Symposium

[4] Consultative Committee for Space Data Systems. (2003). " Space Packet Protocol" CCSDS 133.0-B-1 Blue Book

[5] "ISO/IEC 20922:2016 Information technology -- Message Queuing Telemetry Transport (MQTT) v3.1.1". iso.org. International Organization for Standardization. June 15, 2016.