Design and Implementation of Data Models And Instrument Scheduling of Satellites in a Space Based Internet Emulation System

By

Karthik N Thyagarajan

B.E., Electronics & Communication Engineering

University of Madras, 1999

Submitted to the Department of Electrical Engineering and Computer Science and the Faculty of the Graduate School of the University of Kansas in partial fulfillment of the requirements for the degree of Master of Science

Professor in charge ______________________________

____

____

Committee Members

____

Date Thesis Accepted

To my parents

Without whose love and guidance this would not have been possible

Acknowledgments

First and foremost, I would like to thank my advisor Dr. Gary Minden for giving me an opportunity to work on SBI as well as the RDRN project. It has been a memorable experience working under his guidance and support. I would also like to thank Dr. Victor Frost and Dr. Joseph B Evans for being on my thesis committee.

I wish to thank Leon S. Searl who has taught me a lot about Linux and Networking. He has always helped solve all my difficult queries and his useful suggestions form an important part of this thesis. I also wish to thank all the members of the SBI lab, Pooja, Sujit and Sandhya whose critical comments and suggestions have been of great help to me in my work, Dan Depardo and Artur Leung for all their help on the RDRN project. I would like to thank all my friends out here at KU for always being there for me and for making my stay memorable.

I would like to thank NASA officials who had sent me valuable information when they were approached by email regarding specifications of satellite instruments

Finally, I would like to thank my family for their endless love and support.

Abstract

Earth-observing systems (EOS) are used to study the clouds, water and energy cycles; oceans; chemistry of the atmosphere; land surface; water and ecosystem processes; glaciers and polar ice sheets; and the solid Earth. It consists of a series of polar-orbiting and low-inclination satellites for long-term global observations of the land surface, biosphere, solid Earth, atmosphere, and oceans, which will expand our perspective of the global environment and climate.

Space Based Internet (SBI) aims at applying mobile wireless network technology to satellite systems using innovative topology and routing algorithms. SBI is based on designing a prototype based on this concept and proposes to implement an emulation system to test this prototype. The emulation system shall model satellites in an actual satellite system.

Satellite instruments gather data or make measurements depending on the topology or the region of the earth above which the satellite is or the portion of the earth where the satellite is. This necessitates building a realistic traffic model and scheduling the satellite instruments. Actual satellite systems are modeled to create an accurate and realistic emulation system. This thesis describes the design, implementation of the satellite data models and satellite instrument scheduling based on current satellite systems. Data models represent the region or type of data gathered by satellite instruments and scheduling represents the time duration when the data is gathered.

Table of Contents

List of Figures

List of Tables

Chapter 1

1 Introduction

1.1 Earth Observation Systems

NASA has been studying the Earth and its changing environment by observing the atmos phere, oceans, land, ice, and snow, and their influence on climate and weather. The key to gaining a better understanding of the global environment is exploring how the Earth's systems of air, land, water, and life interact with each other. This approach -- called Earth System Science -- blends together fields like meteorology, oceanography, biology, and atmospheric science. Earth Science Enterprise[1] was established by NASA in 1991 which hoped to expand our understanding of ho w natural processes affect us, and how we might be affecting them by using satellites and other tools to intensively study the Earth.

The Earth Observing System (EOS) [2] is the centerpiece of NASA's Earth Science Enterprise (ESE). It consists of a science component and a data system supporting a coordinated series of polar -orbiting and low inclination satellites for long-term global observations of the land surface, biosphere, solid Earth, atmosphere, and oceans.

Landsat 7[3] was launched in April 1999. Landsat 7 carries a single instrument, the Enhanced Thematic Mapper Plus, which makes high spatial resolution measurements of land surface and surrounding coastal regions.

The Terra[4] satellite (formerly EOS AM -1) provides key measurements that will significantly contribute to our understanding of the t otal Earth system. The Terra instrument complement obtains information about the physical and radiative properties of clouds, air -land and air -sea exchanges of energy, carbon, and water, measurements of trace gases, and volcanology, vegetation, and ocean p hytoplankton.

Several additional spacecraft and instruments are scheduled to fly as part of the EOS program in the next few years.

The Enterprise is currently defining concepts for science and applications missions in the post-2002 time frame. NASA has obtained a first round of ideas from the science and applications communities for these mission concepts and is using them to build a multi-mission profile for Earth observation satellite missions in the 2003-2010 time frame.

Scientists need long-term, consistent measurements of the key physical variables that define the shifts in state and variability of the Earth system components. Lacking these measurements, predictions of the complex responses of the Earth system to human activities and natural variations lack an adequate basis for developing and validating global predictive models and for establishing a baseline to determine trends. Space-based observations hold the key to making progress, because satellites best capture a consistent, global perspective.

1.2 Satellite Data Gathering

EOS satellites have instruments aboard them, each of which gathers data at a specific region below the satellite path on the earth's surface or atmosphere. Depending on whether the region below is land surface, atmosphere, oceans, ice or biosphere the appropriate instrument aboard the satellite turns on or off. As an example, *Terra* satellite has 5 instruments aboard among which *CERES*, *MISR* and *MOPITT* take measurements over the Earth's atmosphere, whereas *ASTER* and *MODIS* take measurements over the Earth's land and ocean.

1.3 Satellite Communications

EOS satellite instruments gather data / measurements over the appropriate region of the Earth's surface. The data gathered by these instruments has to be downlinked to the ground station below.

The satellite has to be in line of site with its ground station to transmit the data. This means the satellite should have high capacity data recorders on it to record the data that is obtained before the satellite is in line of sight with the g round station. The satellite can send data to relay satellites such as the *Tracking and Data Relay Services Satellites* (TDRSS) [5] whose sole purpose is to act as relay satellites between the satellites and their ground stations . This means the EOS satellite must be in line of sight with the relay satellite to be able to send data.

1.4 Problems

Current satellite communications have some disadvantages:

The data collected by the satellite when not in line of sight with the ground station or relay satellite needs to be stored in high capacity solid state recorders. The satellite equipment must be capable of withstanding radiation, high and sudden temperature variations, which might otherwise destroy the data. The entire above poses an increase in cost.

The satellite instrument data cannot be downlinked in real time if there is no line of sight with the ground station. The data might lose its value in cases like tornado or thunderstorm forecasts (natural disasters) if not available immediately.

1.5 Solution – The SBI Approach

Space Based Internet (SBI) aims at applying mobile wireless technology and innovative topology and routing algorithms suitable for satellite systems to enable routing between satellites or satellites and ground sta tions[6]. Actual satellite software is too complex and expensive.

SBI approach is that

- Each satellite shall be capable of generate and switch traffic traveling between other satellites and the ground.
- SBI is scalable – adding a satellite adds capacity to SBI
- Special SBI satellites can be constructed and deployed

SBI shall have two main satellite types:

One of the satellite types has many instruments aboard and is capable of generating and routing data. The other satellite t ype is for the only purpose of switching traffic. An example of this type in a real satellite environment is the *Tracking and Data Relay Services Satellites (TDRSS)*

Satellite Tool Kit (STK) [7] is used to display the emulation environment and is also used to obtain data regarding the satellite's position coordinates, line of sight information with other satellites. STK is a tool from Analytical Graphics Incorporated (AGI).

NetSpec[8] is a network pe rformance analysis tool developed by The University of Kansas and shall be used to generate the various satellite data types and traffic.

1.6 Motivation

The subsections below give a brief idea of the challenges and the solutions in this thesis involving the *Design and Implementation of Data Models and Instrument Scheduling of Satellites in a Space Based Internet Emulation System*

1.6.1 Challenges

To emulate the instrument scheduling and for modeling satellite data the following aspects must be considered:

- Realistic traffic modeling of current / planned satellites
- Obtaining a global topology database
- Realistic scheduling and data generation of satellite instruments in an emulation environment

1.6.2 Realistic traffic models of satellites

Satellites may have one or more instruments aboard them. Each instrument may collect data over a different topology of the Earth. It is therefore essential to have a realistic traffic model for emulating actual satellites. This thesis provides a solution to this by modeling the satellite instrument traffic using actual satellite instrument data models. The information about the satellite instrument specifications has been obtained by researching on EOS satellites.

1.6.3 Obtaining a global topology database

Satellite instruments turn on over the required topology and turn off at other times. In order to emulate satellite links it is essential to emulate instrument-gathering data, which shall be routed by another satellite or sent to the ground station. Therefore it is necessary to emulate satellites turning on and off over the appropriate terrain or surface of the Earth. Therefore, a global topology database was necessary to get topology information given a pair of satellite position coordinates.

The database has been obtained by processing a world map in JPEG format using the JPEG libraries in C and interpreting the RGB values.

1.6.4 Scheduling satellite instruments

Scheduling of satellite instruments is based on their data traffic models. The emulation of instruments generating data and network performance evaluation is done by invoking NetSpec, which is a tool, developed at *The University of Kansas*. Scheduling information for the instruments has been obtained from the NASA web sites. The coordinates of satellite position during emulation time is obtained by using Satellite Tool Kit, tool from AGI

1.7 Thesis organization

Chapter 2 gives a background on EOS satellites. Chapter 3 gives an overview of Space Based Internets (SBI) Concepts and the Emulation System. Chapter 4 discusses how the global Earth surface database was obtained and an analysis of the satellite instrument data that was researched. Chapter 5 discusses the Satellite Data Models developed. Chapter 6 and chapter 7 discuss the design and implementation of the SBI Operations Node Scheduler and SBI Nodes Scheduler respectively. In Chapter 8 the tests conducted have been discussed along with an analysis of results and finally in Chapter 9 conclusions have been drawn based on test results and some possible future work has been suggested.

Chapter2

2 Background on Satellites

Earth Observation Satellites continuously observe the earth from space, and the acquired data are provided as satellite images and are used to study environmental problems, to monitor disasters and to explore resources. A background of satellite observation and data analysis is presented in this chapter[9].

2.1 Earth Observation from Space – Remote Sensing

Remote Sensing is a technology to observe objects' size, shape and character without direct contact with them. The reason why they can be observed without contact because of the characteristics of electromagnetic waves, such as light reflected or radiated from the objects. These reflected or radiated electromagnetic waves are received by sensors aboard earth observation satellites.

In general, the characteristics of reflected or radiated electromagnetic waves depend on the type or condition of the objects. Therefore by understanding the characteristics of electromagnetic waves and by comparing to the observed information, we can know the size, shape and character of the objects.

There are some characteristics of remote sensing by earth observation satellites. Applying these characteristics, the observation data are utilized for various studies related to the earth environment as well as various fields affecting to our lives.

2.1.1 Characteristics of remote sensing by earth observation satellites

- *Enables to observe broad area at a time*

  The observation by satellites can cover a broad area at one time, and it is very useful to find out land use, vegetation distribution, ruins or structures' shape and size. For an example, Japanese Earth Resource Satellite -1 (JERS-1) was used to study the land use and vegetation distribution of Japan.

- *Enables to observe the area for a long period*

  Earth observation satellites orbit the earth repeatedly, so they can observe the same area regularly. Therefore we can see the changes of environment. For example, Marine Observation Satellite-1 observed and gathered images of the area surrounding Mt. Pinatubo on Luzon Island, Philippines which showed land to be covered by greenery before eruption and then with volcanic ash and smoke during the eruption.

- *Enables to know the condition without visiting the area*

  Earth observation satellites observe various areas, so it is useful to find out environmental changes and damage status of the areas where we can not easily go or when a natural disaster occurs. For example, deforestation of the Amazon tropical rain forest in Brazil was investigated using the Japanese Earth Resource Satellite-1 (JERS-1).

- *Enables to know invisible information*

  Observation by satellites can acquire invisible information like temperature, and has provided new discoveries to the earth environmental study. Satellites are used to get temperature distribution over an area when a typhoon spawns. One such satellite is the Marine Observation Satellite 1 (MOS-1)

2.2 Necessity of Satellite Data

The earth observation satellite data are utilized for various studies related to the earth environment as well as various fields affecting to our lives. Some of the valuable information provided by satellites and its data is listed below:

- *Ozone layer depletion*

  Advanced Earth Observation Satellite (ADEOS) continues the series of Total Ozone Mapping Spectrometer (TOMS) total ozone and volcanic sulfur dioxide observations that began with the Nimbus-7 satellite in 1978 and continued through the operation of a TOMS on the Russian Meteor-3 satellite, until that instrument ceased functioning in December 1994.

  Data from another TOMS instrument currently flying on the recently launched NASA TOMS-Earth Probe spacecraft complements the global ADEOS data by providing high-resolution imagery of features related to urban pollution, bio mass burning, forest fires, desert dust and small volcanic eruptions, in addition to ozone measurements.

  In recent years, the depleting effects of industrial chlorofluorocarbons (CFC's) on ozone were demonstrated through the sudden appearance of the Antarctic ozone hole and losses in global ozone. The principle mission of TOMS/ADEOS is to monitor the global ozone trends during the period when CFC-related depletion is predicted to reach a maximum.

- *Land Utilization*

  Landsat 2 land images of Nagoya city have been compared with images 10 years before to find that farms and green lands have clearly been reduced in the suburbs as more downtown or residential areas have been expanded in and around Nagoya and its neighboring places Kariya and Yokkaichi. Moreover other changes around Nagoya, such as forests decrease or coastline changes caused possibly by reclaiming for the past ten years were recognized.

- *Typhoon and rainfall*

  Observation data from the Tropical Rainfall Measuring Mission (TRMM) satellite is used for provision of weather information, such as strength of rainfall, typhoon and thunder, as well as finding causes of global climate changes.

- *Vegetation distribution*

  Information regarding distribution of vegetation is obtained by Vegetation Canopy Lidar (VCL) which gives seasonal vegetation information such as

  - Vegetation distribution for the four seasons
  - Activity of plants according to the seasons
  - Obtain vegetative index which is the amount of vegetation in a particular region

- *Sea surface temperature and phytoplankton concentration*

  Ocean Color and Temperature Scanner (OCTS) provides information regarding the concentration of chlorophyll-a in the sea. It also provides information regarding the ocean temperature.

- *Floating Ice*

  Information about the ice density is obtained by analyzing data from Spot 2 that gives information about the coldest month, the intensity of winter. The satellite was used to study the drift ice of Abashiri that is the southern most tip in the world to have floating ice.

- *Volcanic activity*

  Volcanic activity and damage status can be obtained from satellite observational images and data. For example, the eruption of Mt.Fugendake, Unzen in Shimabara Peninsula, Nagasaki, Japan observed by Earth Observation Satellite, SPOT-2 showed that the pyroclastic flow that hit over the north -eastern slope of Mt.Fugendake, ran along Oshigadani Valley and reached the main stream of Mizunashi River through Kitakamikiba area, after joining the flow formed before. In addition, it showed that the heat wind containing volcanic ashes crossed over Mizunashi River, causing devastating damage to the Ohno-Kiba area.

- *Landforms*

  You can see complicated landforms, characteristic mountain ranges and ruins. Marine Observation Satellite (MOS -1b) was used to study the landforms over Japan

- *Ocean pollution*

  NASDA/EOC has made satellite observations of the oil outflow from the tanker Diamond Sea Grace that took place in Tokyo bay on July 2 1997. The images of oil outflow areas taken from the Japanese Earth Resource Satellite JERS-1 and French earth observation satellite SPOT-2 were used to study and obtain the damage status due to the ocean pollution.

2.3 Remote Sensing Mechanism

Every material on earth shows its own strength of reflection in each wavelength when it is exposed to the electromagnetic waves (visible light and invisible light, such as infrared rays, ultraviolet rays or electric waves). Also, when the material gets hot, it radiates showing its own strength in each wavelength. The strength of reflection and radiation differs according to the condition of objects, such as type of pl ants, status (dead, weak, etc.) or contamination of water.

Sensors aboard the earth observation satellites are capable to acquire the strength of reflection and radiation in each wavelength. From the observation data, we can understand the status of forests, ocean pollution and desertification.

2.4 Orbits of Earth Observation Satellites[10]

The orbit is the path taken by the artificial satellites including earth observation satellites when they fly around the earth. Orbital motion is a common phenomenon in our solar system and elsewhere in the universe -- the Moon orbiting the Earth, the Earth and other planetary bodies orbiting the Sun, stars orbiting each other, and so forth. Few of such orbits are circular, as we assumed above . In general, they are elliptical and follow three very specific rules, or laws.

Kepler's Three Laws of Planetary Motion

Johannes Kepler found that planetary orbits are not circles, but ellipses. Kepler described planetary motion according to three laws. Each of these laws is illustrated by an applet.

<u>Law I:</u> Each planet revolves around the Sun in an elliptical path, with the Sun occupying one of the foci of the ellipse.

Law II: The straight line joining the Sun and a planet sweeps out equal areas in e qual intervals of time.

Law III: The squares of the planets' orbital periods are proportional to the cubes of the semi-major axes of their orbits.

Kepler's laws apply not just to planets orbiting the Sun, but to all cases in which one celestial body or bits another under the influence of gravitation -- moons orbiting planets, artificial satellites orbiting the Earth and other solar system bodies, and stars orbiting each other

Satellites can operate in several types of Earth orbit. The most common orbits for environmental satellites are geostationary and polar, but some instruments also fly in inclined orbits. Other types of orbits are possible, such as the Molniya orbits commonly used for Soviet spacecraft.

There are three types of orbits based on altitude above the earth:

- *Low Earth Orbit (LEO)* – orbits having apogees and perigees below 3000 km. The large majority of all satellites are in Low Earth Orbit.

- *Medium Earth Orbit (MEO)* - orbits having apogees greater than 3000 km but less that 30000 km. These orbits are sometimes used by navigation (e.g., GPS) and communications (e.g., Odyssey) missions.

- *Geostationary Earth Orbit (GEO)* - A satellite in circular orbit around the Earth's equator at 23000 miles altitude (36000 km) will remain stationary over the same location on Earth (i.e., the spacecraft goes around once in its orbit for every revolution of the Earth). This feature is very useful for communications satellites (allowing one satellite to provide continual coverage to a given area of the Earth's surface). As a result, the majority of satellites in geostationary orbit are communication satellites. Some satellites have orbits slightly higher or lower than GEO, but for simplicity sake, all satellites with apogees and perigees between 30000 and 40000 km are termed GEO.

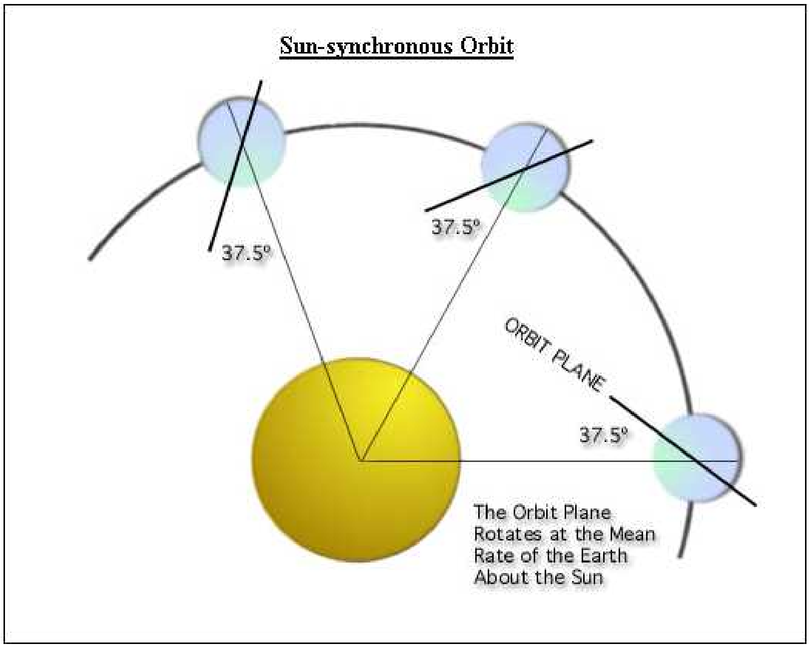
Majority of Earth Observation Satellites generally fall into the category of low earth orbits. There are several types of orbits, and the orbit is selected depending on purposes of the satellites. Earth observation satellites have a major mission to observe the entire earth, so they circle the most suitable sun-synchronous sub-recurrent orbit or a sun-synchronous polar orbit.

Sun-synchronous sub-recurrent orbit is a combination of sun-synchronous orbit and sub-recurrent orbit and sun-synchronous polar sub-recurrent orbit is a combination of sun-synchronous, polar and sub-recurrent orbits.

2.4.1 Sun-synchronous orbit[11]

This orbit is a special case of the polar orbit. Like a polar orbit, the satellite travels from the north to the south poles as the Earth turns below it. In a sun-synchronous orbit, though, the satellite passes over the same part of the Earth at roughly the same local time each day. This can make communication and various forms of data collection very convenient. For example, a satellite in a sun-synchronous orbit could measure the air quality of Ottawa at noon.

Figure 1. Sun-synchronous Orbit



With sun -synchronous orbit, positioning between the satellite and the sun is always the same, which means the area the satellite flies over always gets same sunlight angle. With this orbit, the incidence angle of sunlight to the land surface is always the same, and it allows observing radiation and reflection from the land surface accurately.

2.4.2 Sub-recurrent orbit

A sub-recurrent orbit means that after a certain number of days, the satellite repeats its original orbit. This orbit enables the satellite to observe the same area at regular intervals.

Earth observation satellites utilize the combination of these two orbits, and allow it to observe the same area periodically under the condition of the same sunlight angle to the land surface. In case of Earth Observation-1 (EO-1) satellite, it takes 98.88 minutes to complete a single orbit (this is called the "period"), and the satellite comes back to the same area at almost same time after 16 days (this is the "recurrent period"). Period and recurrent period depend on the satellite.

2.4.3 Polar Orbit

Polar orbit usually has an angle of inclination of 90 degrees to the equator. On every pass around the Earth, it passes over both the north and south poles. Therefore, as the Earth rotates to the east underneath the satellite, which is travelling north and south, it can cover the entire Earth's surface. A polar orbiting satellite covers the entire globe every 14 – 17 days depending on the angle of inclination.

Polar-orbiting satellites provide a more global view of Earth, circling at near-polar inclination (the angle between the equatorial plane and the satellite orbital plane -- a true polar orbit has an inclination of 90 degrees). Orbiting at an altitude of 700 to 800 km, these satellites cover best the parts of the world most difficult to cover in situ (on site). For example, McMurdo, Antarctica, can be seen on 11-12 of the 14 daily NOAA polar-orbiter passes.

These satellites operate in a sun-synchronous orbit. The satellite passes the equator and each latitude at the same local solar time each day, meaning the satellite passes overhead at essentially the same solar time throughout all seasons of the year. This feature enables regular data collection at consistent times as well as long-term comparisons. The orbital plane of a sun-synchronous orbit must also rotate approximately one degree per day to keep pace with the Earth's surface.

2.5 Satellite Data Gathering

The technology used on a satellite varies, depending on its mission. Computers aboard a satellite can receive, store and transmit information in the form of radio signals sent to and from stations on Earth. For Earth scientists, the receipt of the data begins a long process of determining what the data means. By incorporating the data into computer models (which use mathematical formulas called algorithms) researchers can simulate, or model, Earth's processes -- how the atmosphere, oceans and land surfaces interact as a system. Scientists hope that incorporating global satellite data into their computer models will help them better understand the interactive roles of Earth's systems, and help them predict how the Earth's environment will change over time.

Satellite systems provide a high repetition of measurements over the globe, with some systems observing the entire Earth as often as every half-hour. The sensors that are carried on board satellites are individually designed to measure specific physical or biological properties. Satellites are able to observe areas that are not easily accessible from the earth's surface such as in the middle of the vast empty Pacific Ocean. For example, the combination of the global coverage and the high repetition of measurements allow El Nino events to be carefully monitored and allows the link between local El Nino events and wider climatic events to be carefully studied.

There are a total of 24 measurements [12] that can be made by satellite instruments, which is summarized in the table below:

Table 1. Satellite Measurements

#	Region of the Earth	Measurement	Example of Instruments
1	Atmosphere	Cloud Properties	MODIS, GLAS, ASTER
2		Radiative Energy Fluxes	MISR, AIRS, MODIS
3		Precipitation	AMSR-E
4		Tropospheric Chemistry (Ozone)	TES, MOPITT, HIRDLS
5		Stratospheric Chemistry	MLS, HIRDLS, SAGE III
		(Ozone, ClO, BrO, OH)	
6		Aerosol Properties	MODIS, MISR, GLAS
7		Atmospheric Temperature	AIRS, MODIS, TES
8		Atmospheric Humidity	MLS, SAGE III, POSEIDON 2
9		Lightning	OTD
10	Solar Irradiation	Total Solar Irradiance	ACRIM III, TIM
11		Solar Spectral Irradiance	SIM, SOLSTICE
12	Land	Land Cover & Land Use Change	ETM+, MODIS, ASTER
13		Vegetation	ETM+, MISR
14		Surface Temperature	AIRS, AMSR-E
15		Fire	MODIS, ASTER, ETM+
16		Volcanic Effects	MODIS, ASTER, ETM+
17		Surface Wetness	AMSR-E
18	Ocean	Surface Temperature	MODIS, AIRS, AMSR-E
19		Phytoplankton,	MODIS
		Dissolved Organic Matter	
20		Surface Wind Fields	SEAWINDS, POSEIDON 2
21		Ocean Surface Topography	POSEIDON 2, JMR
22	Cryosphere	Land Ice	GLAS, ETM+
23		Sea Ice	POSEIDON 2, ETM+
24		Snow Cover	ETM+, AMSR-E

2.6 Satellite Communications[13]

All satellites need to have some means of communication with Earth; the satellite may need to receive instructions and transmit the information it collects, or it may relay information sent to it to another si te on Earth. This is generally done using some type of antenna. Antennas are defined simply as a piece of equipment that allows transmission and reception of radio signals. Since the information is transmitted using radio waves, which move at the speed of light, this method allows for very fast communications (only a very small time lag)

Satellites gather data and wait to be in line of sight with the ground station to transmit the information collected. Some satellites send information through special relay satellites whose sole purpose is to collect data from other satellites and send it to the ground station. One group of such relay satellites is the Tracking and Data Relay Satellite System (TDRSS).

2.6.1 Tracking and Data Relay Satellite System (TDRSS)

The Tracking and Data Relay Satellite System (TDRSS) [5] is a communication signal relay system which provides tracking and data acquisition services between low earth orbiting spacecraft and NASA/customer control and/or data processing facilities. The system is capable of transmitting to and receiving data from customer spacecrafts over 100% of their orbit (some limitations may apply depending on actual orbit).

The Goddard Space Flight Center receives programmatic direction from NASA Headquarters Office of Space Flight (Code M) regarding all as pects of the Tracking and Data Relay Satellite System and the Space Network. At Goddard, the Mission Services Program Office provides the day to day management and operations of TDRSS and the Space Network.

The TDRSS space segment consists of six on-orbit Tracking and Data Relay Satellites (TDRSs) located in geosynchronous orbit. Three TDRSs are available for operational support at any given time. The operational spacecraft are located at 41, 174 and 275 degrees west longitude. The other TDRSs in the constellation provide ready backup in the event of a failure to an operational spacecraft and, in some specialized cases, resources for target of opportunity activities or dedicated operations.

The TDRSS ground segment is located near Las Cruces, New Mexico and consists of two functionally identical ground terminals collectively known as the White Sands Complex. Customer forward data is uplinked from the ground segment to the TDRS and from the TDRS to the customer spacecraft. Customer return data is downlinked from the customer spacecraft via the TDRS to the ground segment and then on to the customer designated data collection location. TDRSS operational availability exceeds 99%.

Some of the EOS satellites using the services of TDRSS are Upper Atmosphere Research Satellite (UARS), Earth Resources Budget Satellite (ERBS), Landsat, Ocean Topography Experiment (TOPEX), Earth Observing System (EOS) and Tropical Rainfall Measurement Mission (TRMM).

Chapter 3

3 SBI Emulation System

3.1 Drawbacks of current satellite communications

In the case of the current satellites, it is not possible to obtain real time data. The ground station has to wait u ntil the satellite comes into line of sight with it to download the data. This results in storage of the data aboard the satellites using high capacity solid state recorders, which adds to satellite cost.

3.2 Concepts

SBI aims at eliminating the need of using high capacity recorders aboard the satellite and makes satellite data available quickly by:

- Applying mobile wireless network technology to Earth observation satellite systems
- Developing innovative topology and routing algorithms suitable for satellite internet systems
- Building realistic data traffic models of space -based applications of current / planned satellites
- Test the software in an emulation environment

3.3 Emulation

Space Based Internets aims to emulate two point to point links via Ethernet virtual circuits. This shall be done using a managed switch.

3.3.1 Reason for Emulation

Actual satellite network software would be too complex and expensive and so it would be simpler to do emulation.

Advantages of Emulation

- It is less expensive than doing a test on a real satellite network
- It is easier to debug
- It is easier to evaluate the design

Disadvantages of emulation

- It is hard to emulate faster than real time
- It is scalable up to 10s of nodes but will become complicated and unwieldy beyond that

The SBI system shall provide IP over Ethernet facility between satellites and satellites and ground station. Thus it provides a common protocol and eliminates the necessity of special communication channels between the satellites and ground station.

3.4 SBI Satellite Features

The following provide a comparison between the actual satellite features discussed in the previous chapter and the SBI satellites.

3.4.1 Orbits of SBI satellites

SBI nodes that are capable of both generating and routing data shall represent Low Earth Orbit satellites. SBI nodes that are capable of only routing data shall represent Medium Earth Orbit or Geosynchronous satellites. The satellite orbits are sun-synchronous, polar, sub-recurrent orbits, which may be circular or elliptical. The communication resources available between any two nodes in the network vary over time due to the variation and absence of uniformity in the orbits.

3.4.2 SBI Satellite Data Gathering

SBI satellites gather data over the following regions of the Earth:

- Land
- Ocean
- Ice
- Atmosphere
- Solar irradiance through the Earth's atmosphere

3.4.3 SBI Satellite Communications

SBI satellites are capable of communicating directly with the ground station below provided they are in the line of sight. If they do not have line of sight with the ground statio n, they can communicate through satellites with which it has line of sight, which shall act as a router. The router in turn shall have line of sight to another router or the ground station.

Two types of communication links are emulated in SBI:

- Data link

The data link exists between satellites and between satellites and ground stations. The data link serves as the data link between satellites and ground station. There are two data links. One of the links can support up to a total data rate of 100 Mbps. The other link may support up to 1 Gbps. A satellites data link bandwidth is fixed and is capable of routing or sending data only up to the maximum rate. The data is sent over UDP. All data into and out of the SBI nodes shall be in the Internet Protocol (IP).

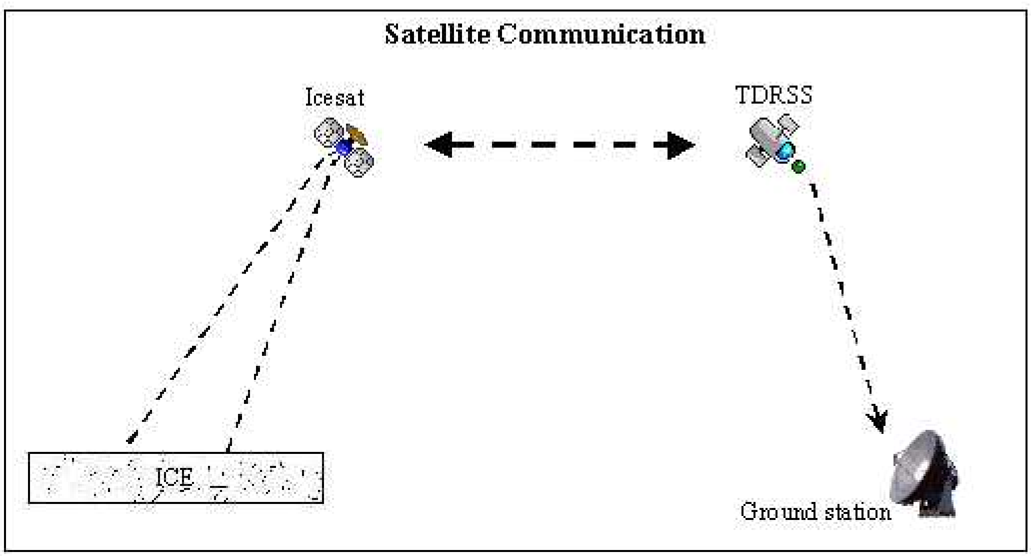
- Management link

The management link exists only between ground station and satellites. It is meant for the ground stations to send management information to the satellite nodes. For example, the ground station will send signals over the management l ink, to the satellites to turn their instruments on or off.

SBI nodes use multiple emulated channels or beams (RF and/or Optical) for communication. The SBI prototype will not emulate steering of antenna beams. It assumes that the beams are steered by some other means.

3.4.4 SBI Node Types

Figure 2. Satellite Communications



SBI nodes can be one among the following:

- Satellites capable of generating and routing traffic

These shall represent the general Low Earth Orbit satellites in an actual environment with an ad ded routing / networking capability. They are capable of generating, forwarding or receiving data. The data rates vary between a few Kbps up to 150 Mbps. These satellites shall have a minimum of two transmitter -receiver pair to enable routing among satellites. The bit rate of the satellite's communication link is sufficient to carry the peak data rate of all instruments on the satellite that can make observation simultaneously. These satellites are generally in LEO at heights between 350 km to 800 km.

- Data Source and Relay Satellites

These are satellites that only route data to other satellites or ground stations. They have high bandwidth capability so that they can route data between many satellites. They only receive or forward data. They are not capable of generating data. An example of this in an actual satellite environment Medium Earth Orbit (MEO) satellites such as the Tracking and Data Relay Satellite Systems (TDRSS). MEO satellites have shorter delays with less number of satellites in line of sight and GEO satellites have more delays but also more satellites in line of sight.

- Facilities or ground stations

These nodes shall represent the ground stations. One such facility or ground station shall also serve as the Operations Node, details of which sha ll be discussed in Chapter 6. The ground station is capable of receiving data over a high bandwidth channel and sends management information to the satellite nodes over a low bandwidth channel. It should be noted that in an SBI emulation system, only the ground station that serves as the Operations Node shall be capable of sending management information to the satellites.

3.5 SBI Emulation System

The SBI emulation system consists of three parts:

- SBI Node Software[14]

  Each SBI node represents a satellite or ground station. The SBI node software can be loaded in actual satellites and ground stations to enable them to route traffic. Each node though representing a single satellite can be used to represent multiple antennae by means of virtual Ethernet (veth) [15], where each veth in a node represents an antenna on the satellite.

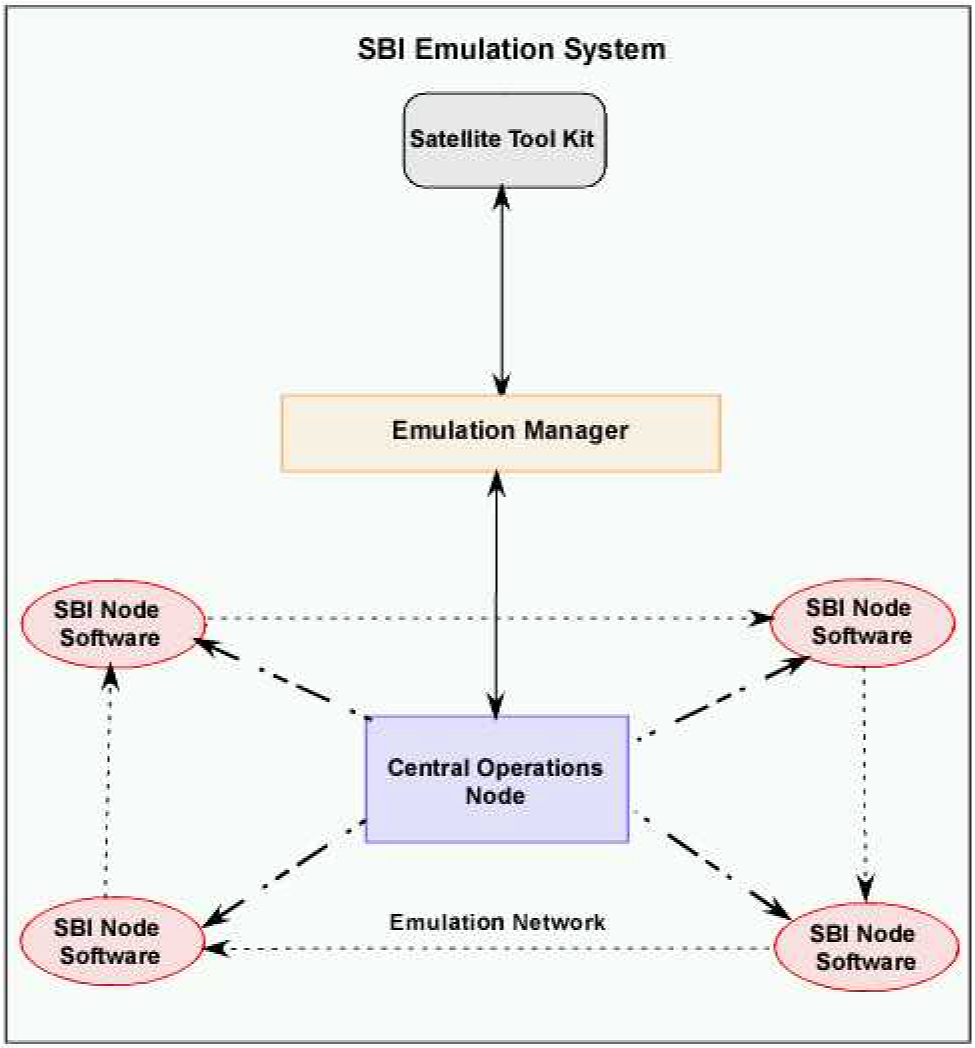
- Emulation Manager[16]

  The Emulation Manager is responsible for the configuration, control and monitoring of the status of the SBI emulation. The EM emulates the actual environment in which the SBI system will be deployed to make the SBI Nodes think that they are operating in a real network. Through the SBI Emulation Software the user is able to create sce narios of SBI nodes and to execute the scenarios. During scenario execution the user controls and monitors the progress of the emulation.

- Emulation Network

  It is responsible for emulating communication links in space with associated bit errors, delay and data rates. It emulates the actual SBI environment in as realistic an environment as possible.

Figure 3. SBI Emulation System



The emulation system will be discussed in more detail under sections 1.6, 1.7 and 1.8

It should be noted that:

- As each satellite is l aunched, it adds capacity to the SBI making the system scalable.
- Special SBI satellites can be designed, constructed and deployed to provide extended capability and capacity
- Communications between satellites shall be necessary for unique events, for coordinated data collection.

3.6 SBI Node Software

The SBI Node Software consists of software that is resident on the SBI nodes that serve as satellites or ground stations. It is the prototype node software that can be deployed in SBI satellites.

The SBI node software is classified into two types:

- Operations Node Software
- Common Node Software

3.6.1 Operations Node Software[17]

Operations node software is the software that resides only on the operations node. The Operations Node determines the IP routes between the SBI network nodes. Its functions include:

- Determination of network topology
- Configuration of the routing tables for all the nodes
- Scheduling data collection for satellite node instruments
- Orbit or ground station location determination

It determines the topology of the network based on node connections. It computes the routing table based on the network topology and transfers it to the other SBI nodes using dynamic routing protocol. It schedules instruments for data collection.

Connections between 2 satellite nodes is based on the following factors:

- Link capacity of the instrument
- Line of sight (LOS)
- Amount of time the satellites would be in LOS with each other
- Propagation delay of the satellite link

The operations node software consists of:

Communication module:

It is the module that connects the Operation Node with the other SBI nodes. The data that goes through the communication module are the routing and scheduling information to the SBI nodes.

Scheduling System:

The schedul ing system is responsible for scheduling of the satellite instruments. The scheduling system consists of the following modules:

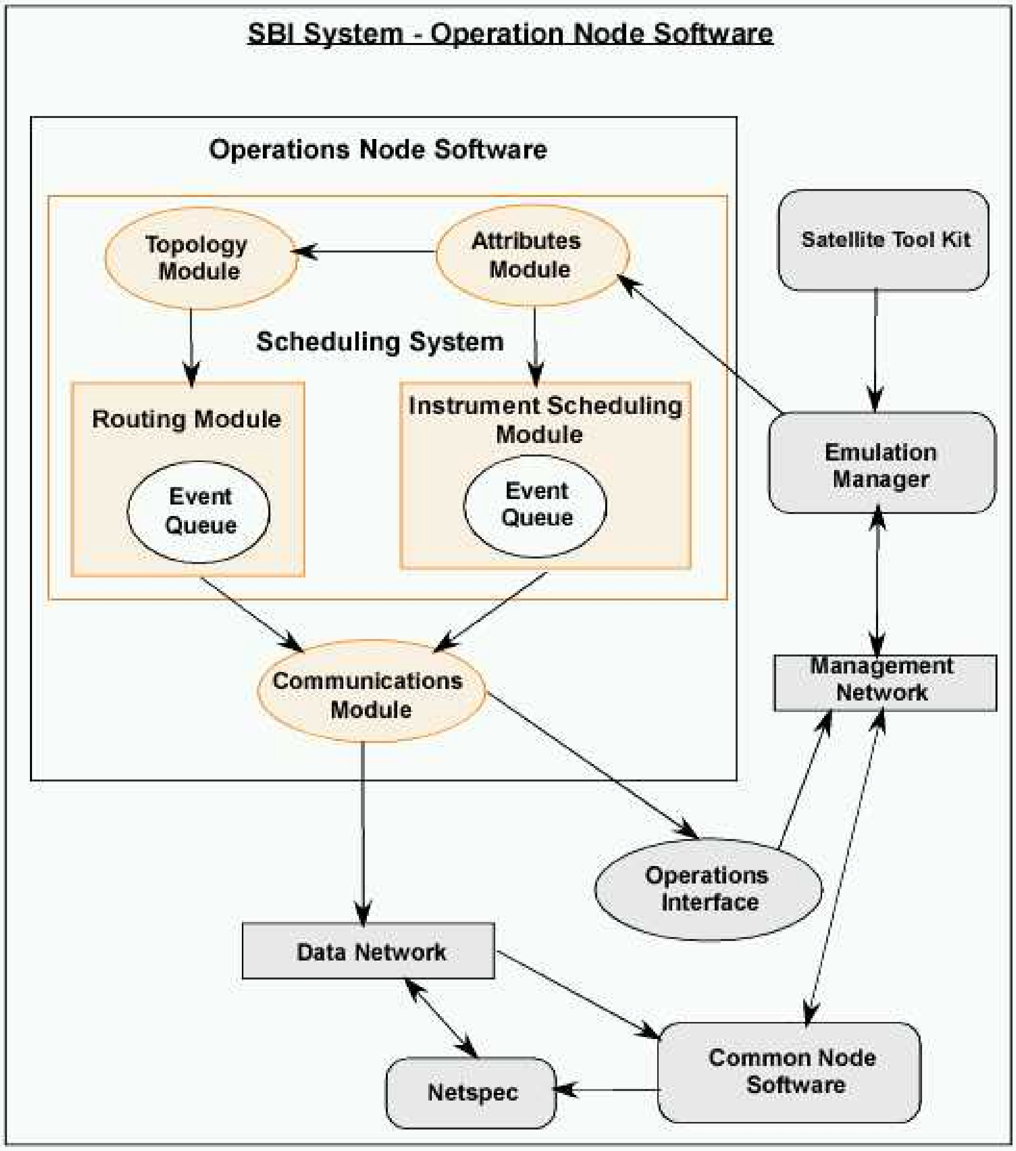
3.6.1.1 Attributes Module

The attributes module consists of satellite attributes such as the satellite position, number of instruments on the satellite, the data characteristic of the instruments such as data rate, burstyness, satellite orbital parameters such as orbital period, altitude and so on. The attributes module receives the above information for all the nodes in the network from the emulation manager. Some of the attribute information is passed onto the topology module.

The node table contains the following information:

- IP address for each node interface. Each node interface represents a virtual device.
- Type of Node : Relay sa tellite node, Source and relay satellite node, ground station
- Orbital parameters of the satellite node.
- Current or future positions of the nodes in terms of latitude, longitude and altitude.
- Communication characteristics: Bit Rates, Signal Strength (Node C apacity), current capacity utilization, Propagation delay.
- Current time using the system clock.
- The number of instruments on the node, which specifies the number of connections that node, can make. For emulation purposes, it represents the number of virtual devices on the node.

Figure 4. SBI Operation Node Software



The updates are obtained time to time from the emulation manager.

3.6.1.2 Topology Module

The topology module obtains the attributes of the SBI nodes in the network from the attribut es module. Based on the attributes the topology module decides on the connections between the nodes. This information is then passed onto the routing module.

The connection decisions are based on the following factors:

- Line of Sight (LOS) between the satellite nodes
- Each data source satellite must be in line of sight with a ground station or a relay satellite to route data
- Distance between 2 satellites
- Bandwidth, propagation delay, duration of the connection

3.6.1.3 Routing Module

The routing module receives the connection updates from the topology module and accordingly stores the routing table changes, which is then put onto an Event Queue to be updated. It maintains the routing tables for all the SBI nodes.

The routing tables are similar to the IP Kernel Routi ng tables. It contains the following important information:

- IP address of the node (so that the routing table is transmitted to the correct SBI node.
- For each entry, it stores the destination IP address, the gateway and the physical interface on which it has to transmits and receive packets.

Initially, the routing table contains entries, which would specify routes only to the next hop. Based on these the routing among the SBI nodes is determined and the changes updated.

3.6.1.4 Operations Node Instrument Scheduling Module[18]

The operations node instrument scheduling module obtains information for each SBI node from the attributes module. It is responsible for scheduling the satellite instruments. The scheduling information is put in a n Event queue, which is then sent to the appropriate SBI nodes through the communications module. The operations node instrument scheduling module will be dealt in detail in Chapter 5.

The Operations interface is discussed in section 1.7 Emulation Manager . Satellite Tool Kit (STK) and NetSpec are discussed in section 1.8 Other Modules

3.6.2 Common Node Software

The common node software is resident on all the SBI nodes and is responsible for the following functions:

- Routing table updates
- Scheduling satellite instruments

It receives routing -table information as per the decisions made by the Central Operations Node and updates the node's kernel routing tables. Based on the commands obtained from the operations node it schedules the satellite instrument to transmit data.

In a SBI network scenario, the SBI nodes establish connection in order to switch IP traffic between them. The decision for establishing a connection between 2 SBI nodes is made by the central operations node. This decision is then conveyed to the SBI nodes, which transmits that information to the Emulation Manager.

The common node software consists of the following modules:

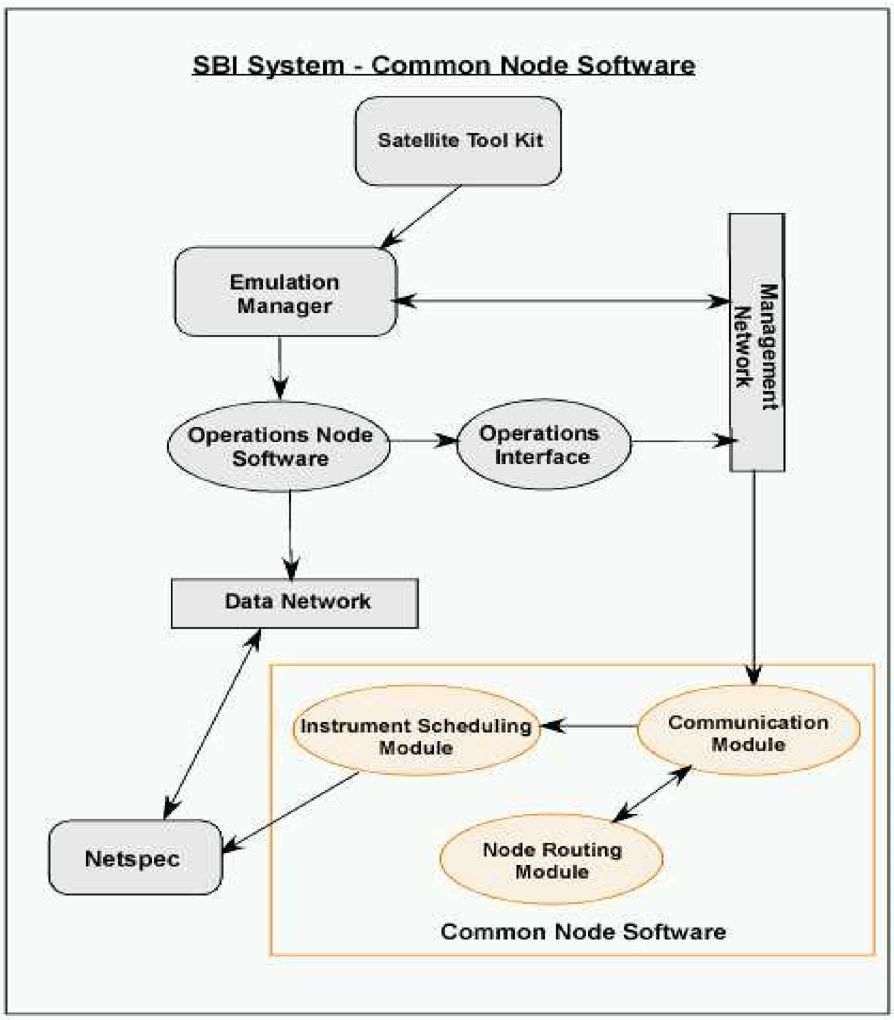
3.6.2.1 Communications Module

This module is responsible for communications that take place with the operations node such as routing t able updates and scheduling information. The communications module obtains the data and sends it to the appropriate module.

3.6.2.2 Node Routing Module

This module obtains the routing table updates from the operation node through the communications module. It updates its routing table based on it.

Figure 5. SBI System – Common Node Software



3.6.2.3 Node Instrument Scheduling Module[19]

The node instrument scheduling module obtains the scheduling information from the operations node through the communications module. It is responsible for scheduling the instruments based on the scheduling data obtained. The node instrument scheduling module will be dealt with in more detail in Chapter 6.

3.7 Emulation Manager

As mentioned previously, the Emulation Manager configures controls and monitors the status of the SBI emulation. The EM emulates the actual environment in which the SBI system will be deployed to make the SBI Nodes think that they are operating in a real network.

Emulation consists of:

- Nodes emulating SBI capable satellites or ground stations
- Orbital data of satellites and the position of satellites and ground stations
- Data link information emulating the payload and communication capability of satellites
- Emulation environment topology
- Specific payload instrument data generation models that are used in the emulation to test and evaluate SBI configurations in the laboratory under controlled and repeatable conditions.

The Emulation Manager functions as the node-controlling unit and acts as an interface between the nodes in the scenario and the user. The node may be one of the following three: a stationary ground station, a satellite in orbit around the Earth or an end user node collecting the data from the ground station.

3.7.1 Emulation Manager

The Emulation Manager consists of

- Node Manager

  All control messages are sent via the management network to the SBI nodes through the node manager. The communication messages received by the node manager are sent to the EM Controller

- Operations Channel Manager

  The operations channel manager is responsible for message transfer between the operations node and the other SBI nodes. The messages transferred are scheduling information or routing updates and network topology information.

- Connection Manager

  The Connection Manager Module interfaces with the Managed Ethernet Switch and controls the connections among the different SBI Nodes. Commands for creating and removing connections between the nodes are sent out through this module. On receiving a connect request between 2 SBI nodes, the connection manager verifies if they are in line of sight with each other before establishing a connection. On receiving a connection termination request, the connection manager again receives this information from the Node Emulation Software and physically removes the connection on the managed Ethernet Switch.

The other modules of the emulation manager are briefly discussed below:

- Event Queue: It contains a list of events that will occur, for example, the loss of LOS between satellites that are connected.
- Event Manager: It is responsible for event scheduling on all the components of the SBI Emulation System. The Event Manager sends out commands such as start, stop, restart, pause and resume to the Nodes through the Node Manager. The Event Manager also performs operations related to time keeping.

- Orbital manager contains various modules that have important functions such as orbital position and propagation calculations (Orbital Calculator), access time computation, LOS times (Statistical Calculator), node database with details on type of orbit, mission and other such information.
- User Interface Manager implements requested configuration changes, reports current state as requested by the user. The I/O Module performs the operations related to the user interface displays. The Config Module retrieves data from the Config files regarding all the attributes of the currently executing scenario and routes it to the Orbital Manager. This information is stored in the Node Database. The Log Module writes out log data periodically to the log files corresponding to events occurring in the emulation. This data may be used by the Statistical Calculator module to compute statistical calculations.

3.7.2 Node Emulation Software

It is the software that is resident on the SBI nodes. It starts up the node software on a command from the Emulation Manager. It consists of the Node Controller and Communication Unit.

The Node Controller consists of:

- Node Control controls the other emulation software modules. It acts on any command that it may receive from the emulation manager such as pause, start, etc. It also does node monitoring functions
- Manager Interface is responsible for sending the orbital data to the node software, besides doing node monitoring functions. Commands such as start, stop, etc. are received by this module from the EM and routed to the Node Control.

- Communications controller is responsible for the communication interfaces. Multiple communication channels are implemented in the nodes by using virtual Ethernet devices for the communication. The bit errors, data link rate and delay for the communication links are retrieved from the EM by this module and sent to the Communication Interface.
- Routing Stub module intercepts changes to the Routing Table on the nodes and sends the updated changes to the Emulation Manager through the Manager Interface. The EM acts on this data by adding or removing a connection between nodes on the managed Ethernet Switch.

The communication unit consists of:

- Connection Interface acts as the interface between the nodes and is connected to the Managed Ethernet Switch. Data transmission, reception and forwarding is performed by this module.
- Operations Interface acts as the transceiver interface for the reception of commands from the Operations Node via the unmanaged Ethernet Switch.
- Routing Table contains entries corresponding to the best route for data flow on the SBI network.

3.7.3 Operations Interface

The mechanism that should always allow communication with a SBI node is the Operations Interface of the Emulation Software. The Operations Interface represents a communication medium like the S-Band command channel used to control many present day satellites. All routing and scheduling information go to the SBI nodes through the operations interface.

3.8 Other modules

Other modules that are used in SBI are NetSpec and Satellite Tool Kit (STK)

3.8.1 NetSpec[8]

NetSpec is a network level end -to-end performance evaluation tool developed by researchers in the University of Kansas, to help in the collection of results delivered by performance experiments on the ACTS ATM Internetwork (AAI) project. The *NetSpec* system provides support for large scale data communication network performance tests with a variety of traffic source types and modes. This software tool provides a simple block structured language for specifying experimental parameters and support for controlling performance experiments containing an arbitrary number of connections across a LAN or WAN.

NetSpec exhibits many features that are not supported by the most often performance tools used today (ttcp, Netperf) like:

- Parallel and serial multiple connections
- A range of emulated traffic types (FTP, HTTP, MPEG, WWW) on the higher levels
- The most widely transport protocols used today, that is TCP and UDP
- 3 different traffic modes (full stream, burst, queued burst)
- Scalability
- Ability to collect system level information from the communicating systems as well as intermediate network nodes.

NetSpec is a tool that was designed to simplify the process of doing network testing, as opposed to doing point to point testing. NetSpec provides a fairly generic framework that allows a user to control multiple processes across multiple hosts from a central point of control. NetSpec consists of daemons that implement traffic sources/sinks and various passive measurement tools. It is a tool designed to provide convenient and sophisticated support for experiments testing the function and performance of networks

NetSpec uses a scripting language that allows the user to define multiple traffic flows from/to multiple computers. This allows an automatic and reproducible test to be performed.

NetSpec supports three basic traffic modes:

- **Full Stream Mode**
  This mode is otherwise called full blast mode where it instructs the test daemons to transmit data as fast as possible.

- **Burst Mode**
  In this mode each time the timer interrupt expires, the interrupt handler writes out the blocks. The user also specifies the burst period and the burst size, in bytes/block and blocks/burst. The drawback of this algorithm is that if a write call initiated in response to one interrupt does not finish before the next interrupt occurs, it will fail to write out block(s) for the second interrupt.

- **Queued Burst Mode**

  This mode is a variation of the basic burst algorithm, which separates the interrupt service process from the block writing process with a queue. The user specifies the same parameters as in the burst mode. The advantage of this algorithm is that variations in available line rate will not cause it to miss blocks generated by interrupts arriving before previous write completes. The drawback is that characteristics of the traffic is influenced by the queuing delay

Emulated traffic Modes

NetSpec has the potential to emulate FTP, TELNET, VBR Video Traffic (MPEG, Video -Teleconferencing), CBR Voice Traffic, and HTTP traffic (World Wide Web traffic) on the application layer. This feature makes NetSpec one of the unique Network Performance Evaluation Tools supporting such kind of traffic types and comes out to be an essential tool for the Network Engineer, being able to test the network's performance under different emulated traffic types.

Example NetSpec script:

In the script below, the type of traffic characteristic is burst. In the burst traffic mode, the hosts under test transmit data every some specific intervals, specified by the burst period as shown in the script above. The burst size and the burst period is passed as a parameter by the user. This mode is very useful in real world experiments where rate mismatches might reduce the throughput dramatically.

The first block specifies the characteristics of the traffic source, while the second block specifies the characteristics of the receiver host. The sender and receiver TCP systems used are machines in the local ATM network in the University of Kansas (KU). This script can be used in testing the performance in a LAN and a WAN, as well as testing the performance over a satellite link, where traffic shaping is necessary due to link mismatches. The traffic in this script is shaped to 128 Kbps.

Throughput is calculated as:

$$\textbf{Throughput} = (\textbf{blocksize} * 8) / (\textbf{period} * 10^{-6}) \ \textbf{bits per second}$$

Point-to-Point traffic shaped at 128 Kbps

```
cluster
{
	test 192.168.10.2
	{
		type = burst (blocksize = 160,period = 10000, duration = 105);
		protocol = tcp (window=1048576);
		own = 192.168.10.2:45000;
		peer = 192.168.10.1:45001;
	}

	test 192.168.10.1
	{
		type = sink (blocksize = 160, duration = 105);
		protocol = tcp (window=1048576);
		own = 192.168.10.1:45001;
		peer = 192.168.10.2:45000;
	}
}
```

cluster:

This indicates the beginning of a structured block, where the source and destination end systems are defined, as well as the various network parameters. If a machine domain name or its IP follows (after) then the control daemon (nscntrld) will run on that machine. In the example, nscntrld runs on the machine with IP address 192.168.10.2

test:

This specifies where the test daemon (nstetsd) will run. For th e example, nstestd shall run on machines 192.168.10.2 and 192.168.10.1

type:

It indicates the mode in which the test will be run. In this case the mode is burst.

blocksize:

This indicates the size of blocks in bytes

duration:

It indicates the duration of the experiment in seconds

period:

It represents the burst period in microseconds. It is used only in burst and queued burst modes.

protocol:

Indicates the transport protocol used in the test. In this case the protocol used is **TCP** and so it has a **window** which indicates the window size. If UDP is the protocol used, the window size is not given.

own:

Indicates the local machine by domain name or IP address.

peer:

Indicates the local machine (by domain name or IP address) where the test daemon is running within the block.

Appendix contains NetSpec scripts and their outputs with explanations.

3.8.2 Satellite Tool Kit version 4.2 (STK 4.2)[7]

SBI makes use of STK for emulation and for obtaining data. STK is used to emulate the satellite environment and its data are used for obtaining information to schedule instruments. Satellite Tool Kit (STK) is the core in a suite of analysis software tools that addresses all phases of a space systems life cycle, from policy development and design to launch and operations.

Designed to provide sophisticated modeling and visualization capabilities, STK performs functions that are critical to all mission types. These functions include propagating vehicles, determining visibility areas and time s, and computing sensor pointing angles and generating results in textual and graphical formats. The STK user interface simplifies and streamlines these functions for novice and expert users alike.

Based on simple user inputs, STK generates paths for a variety of space- and ground-based objects, such as satellites, ships, aircraft and land vehicles. STK provides animation capabilities and a two-dimensional map background for visualizing the path of these vehicles over time. STK can model the field of view available to these objects, enabling the user to determine whether the object can "see" its target. STK's extensive reporting and graphing capabilities provide quick computations of accesses and times, position and other key information, helping users to communicate results effectively.

SBI makes use of 3 modules in STK:

- STK / VO

  The 3-D VO window allows viewing of complex information in a visually pleasing yet technically accurate environment. Using the 3-D VO window, it is possible to study space systems directly by observing space, air, land and sea object relationships as they exist in the past, present and future. Flexible scene-viewing controls allow creation of informative "fly-through" sequences showing scenario-specific features. The VO window displays information about the scenario in 3-D. A number of VO window display options can be specified directly from an STK/VO session, including globe preferences, celestial attributes, display options, model preferences, and more.

- Connect

  STK's Connect module provides an easy way to connect with STK and work with it in a client-server environment. The library shipped with Connect can be used to easily set up and use TCP/IP or domain sockets. This library contains functions, constants and other messaging capabilities that can be used to connect third-party applications to STK. With Connect, optional diagnostic messages can be generated. Additionally, Connect allows overriding the standard messaging and modifying it or use of own messaging format for compatibility with third-party applications. These features allow for better control the messaging environment. Connect links with STK and STK/VO so that it is possible to visualize events in real time. For instance, Connect can be used to feed real-time telemetry data from the launch and early orbit of a mission. As a scenario, the data can be viewed in 2-D or 3-D to simulate the mission and assist in understanding and resolving any issues that may arise.

- Chains

  The STK Chains module is a multi satellite, multi target/ground station productivity tool with powerful analysis capabilities that aren't available in any other off-the-shelf package. The Chains module allows extension of the pair-wise analysis capabilities of STK to include accesses to and from satellite constellations, ground station networks, groups of targets and multiple sensors. For instance, the time period in which a satellite can see a target and a relay station and the relay station can see a ground station can be determined. With Chains, objects can be grouped together—a powerful capability that allows checking accesses to the entire group. For example, it is possible to determine when a tank has access to at least four satellites in the Global Positioning System (GPS) constellation.

In addition, it can be easily determined when LandSat can see a target as well as a TDRS relay satellite, and the relay satellite can, in turn, see a ground station. More sophisticated problems can be solved such as allowing the use of TDRS -East or -West as the relay satellite. In Chains, a constellation of relay satellites is defined and then incorporated into a chain.

Chapter 4

4 SBI Satellite Data Models

4.1 Necessity of data models

Satellite instruments gather data or make measurements depending on the topology or the region of the earth above which the satellite is or the portion of the earth where the satellite is. This necessitates building a realistic traffic model, which aids in scheduling the satellite instruments. Actual satellite systems are modeled to create an accurate and realistic emulation system.

An analysis of the satellite and instrument behavior has been done and the following sections describe the conclusions reached about the satellite data models. A total of 24 EOS satellites and their instruments (total of 68) have been studied.

4.2 EOS satellites' orbit characteristics

Among the 24 EOS satellites studied the orbit characteristics were as follows:

Table 2. EOS Orbit Characteristics

#	Satellite	Period	Altitude	Orbit
		minutes	km	
1	EO 1	98.88	705	sun synchronous, circular
2	Lightsar	97.6	600	sun synchronous, circular
3	SRTM	89.1	230	elliptical
4	VCL	92.56	400	circular
5	Terra	98.88	705	Sun synchronous, polar
6	Poseidon-1	113	1366	circular
7	Poseidon-2	112	1344	circular
8	Seastar	98.88	705	sun synchronous
9	Seawinds - 1A	100.87	800	elliptical
10	ADEOS	99.65	830	sun synchronous
11	ADEOS II	101	802.9	sun synchronous
12	QuikSCAT	100.6	800	elliptical
13	Cloudsat	98.88	705	sun synchronous
14	Picasso-Cenna	98.88	705	sun synchronous, circular
15	ICESAT	97.6	600	non-polar, leo
16	UARS	96	575	near-circular
17	EOS PM	98.88	705	sun synchronous, polar
18	EOS-Chem	102	705	sun synchronous, polar
19	Jason 1	112	1336	Circular
20	ML1-OTD	100	710	near polar
21	SPARCLE	90.52	300	low earth, polar
22	TRMM	92	350	non sun synchronous, circular
23	SAGE II	96.8	650	non sun synchronous, circular
24	SAGE III	105.55	1020	sun synchronous, circular

It is observed that all of the 24 satellites (68 Instruments) studied have Low Earth Orbits. A primary advantage of LEOs is that the transmitting terminal on earth doesn't have to be very powerful because of the low orbit. LEOs also are smaller than the large GEOs, less costly, and cheaper to launch. The propagation delay is considerably less and there is limited station keeping expense. Flying in a Low Earth Orbit, enables better resolution of the Earth below, more revolutions around the Earth and thereby more number of downlinks with the ground stations below.

It should be observed that TDRSS (Tracking & Data Relay Satellite System)[5] has not been included here as part of EOS satellites, though it is a part of SBI. TDRSS provides continuous global coverage of Earth-orbiting satellites at altitudes from 750 miles to about 3,100 miles. At lower altitudes, there will be brief periods when satellites or spacecraft over the Indian Ocean near the equator are out of view. Receive data rates are 300 megabits/second at Ku- and Ka-band, and 6 Mbps at S-band. The spacecraft carries additional capability for Ka-band receive rates of up to 800 Mbps. Transmit data rates are 25 Mbps for Ku- and Ka-band, and 300 kilobits/second for S-band.

The fully operational TDRSS network consists of three satellites in geosynchronous orbits. The first, positioned at 41 degrees west longitude, is TDRS-East (TDRS-A). The next satellite, TDRS-West is positioned at 171 degrees west longitude. The remaining TDRS are positioned above a central station just west of South America at 62 degrees west longitude as a backup. The satellites are positioned in geosynchronous orbits above the equator at an altitude of 22,300 statute miles. At this altitude, because the speed of the satellite is the same as the rotational speed of Earth, it remains fixed in orbit over one location. The positioning of two TDRSs will be 130 degrees apart instead of the usual 180-degree spacing. This 130-degree spacing will reduce the ground station requirements to one station instead of the two stations required for 180-degree spacing.

The TDRS system serves as a radio data relay, carrying voice, television, and analog and digital data signals. It offers three frequency band services: S-band, C-band and high-capacity Ku-band. The C-band transponders operate at 4 to 6 GHz and the Ku-band transponders operate at 12 to 14 GHz. TDRSS also provides communication and tracking services for low Earth-orbiting satellites. It measures two-way range and Doppler for up to nine user satellites and one-way and Doppler for up to 10 user satellites simultaneously. These measurements are relayed to the Flight Dynamics Facility at Goddard Space Flight Center (GSFC) from the White Sands Ground Terminal (WSGT).

4.3 Data Rates

The table and the graph below shows the data rate ranges for EOS satellites.

It can be observed that the data rate of a predominant number (> 85%) of satellite instruments are in the range of 1 Kbps – 100 Mbps.

Table 3. EOS Satellite Instrument Data Rates

Data Rate		#
From	To	Instruments
100 bps	1,000 Kbps	6
1 Kbps	10 Kbps	16
10 Kbps	100 Kbps	14
100 Kbps	1 Mbps	14
1 Mbps	10 Mbps	8
10 Mbps	100 Mbps	7
100 Mbps	1 Gbps	3

Most of the satellite instruments have a bursty data gathering characteristic. Some of the satellite instruments have pictures or images of various parts of the Earth. Such instruments have MPEG type data.

4.4 Data type or Characteristic

Most of the satellite instruments have a constant bit rate, bursty traffic characteristic. A few instruments such as WFC (Wide Field Camera) aboard Picasso - Cenna transmit images or pictures and so their data characteristic is MPEG traffic.

4.5 Satellite Instrument Data Models

EOS satellite instruments make measurements based on the region of the Earth in their view, or the Earth's atmosphere. The region of the Earth is based on the topology below on the Earth's surface or atmosphere.

4.5.1 Classification

Based on the region of measurements done by the satellite instruments the data models can be classified as follows:

- Earth Surface (ES) measurements
- Solar Occultation or Sunrise-Sunset (SS) measurements
- Day – Night (DN) measurements

Among the 68 satellite instruments studied, 28 instruments have observations based on a particular Earth surface type below it.

Table 4. Satellite Instrument Measurements

Measurement	# Instruments
Earth Surface Type	42
Sunrise-Sunset	6
Day / Night	20

14 other instruments observe all surface types of the Earth. Advanced Land Imager (ALI) of EO1 satellite takes measurement only over land, Poseidon -1 takes measurements only over the oceans and GLAS of Icesat observes only ice. 20 instruments make measurements only during satellite day. Solar occultation measurements are made by 6 instruments (e.g. ILAS II). 29 instruments among the 68 measure the atmosphere over the Earth surface.

4.5.2 Earth Surface Measurements

The Earth's surfac e may be classified as Land, Ocean and Cryosphere. The other measurement is through the Earth's atmosphere.

Measurements over the land are made to analyze factors like soil moisture (AMSR), desertification and deforestation studies (AVNIR), distribution of vegetation and urban area (AVNIR), and to create detailed maps of land surface temperature, emissivity, reflectance, and elevation (ASTER). Land measurements can be over specific vegetative regions or arid land regions.

Measurements over the ocean are used to determine weather conditions, ocean surface temperature, studying wind patterns (seawind) and study of biological productivity of ocean waters by measuring concentration of chlorophyll (MISR).

Albedo values can be obtained in the Cryosphere. In the climatologically important Polar Regions, albedos of the ice -and-snow-covered areas (known as the Cryosphere) are continuously modified by natural processes and by human sources of pollution. This affects the amount of solar energy reflected by the sur face. A "feedback mechanism" between the atmosphere and the Cryosphere results -- if a snow-covered surface is partly blackened by soot particles, for example, the surface will absorb more solar energy, which may melt the snow, and darken the surface further.

Conversely, if a surface is whitened by a deposit of fresh snow, it will reflect more sunlight than before, helping to preserve the snow cover (MISR). MODIS maps the areal extent of snow and ice brought by winter storms and frigid temperatures.

Atmospheric measurements are done to provide better estimates of the amounts of aerosols and gases fires release into the atmosphere (MODIS), ozone levels (TOMS). TES measurements provide better understanding of long-term variations in the quantity, distribution, and mixing of minor gases in the troposphere, including sources, sinks, troposphere-stratosphere exchange, and the resulting effects on climate and the biosphere.

In the following table and chart the following abbreviations are used:
L – Land
O – Ocean
I – Ice / Cryosphere

The tables below give the distribution of instruments with respect to the Earth surface type.

Table 5. Surface Specific Measurements

Surface specific measurements	
Measurement	**# Instruments**
Only L	7
Only O	6
Only I	1
L & O	13
L & I	1
O & I	0
L, O, I	14

4.5.3 Sunrise Sunset Measurements

The solar occultation measurement technique is a very simple method of measuring vertical profiles of atmospheric optical depth profiles from Earth orbit using the sun as a light source. As the spacecraft orbits the Earth, the satellite instrument points toward the sun and measures its intensity. It observes sunsets when the spacecraft moves from the sunlit toward the dark side of the Earth.

Before each sunset starts, the line-of-sight (LOS) between the spacecraft and the sun is unobstructed by the atmosphere so that the sun's intensity as measured by the instrument is unattenuated. But, when the spacecraft starts to dip below the horizon so that the LOS passes through a portion of the atmosphere, the sun's intensity will be attenuated due to aerosols and gases in the atmosphere that scatter and absorb sunlight. During sunrise events, when the spacecraft moves from the dark towards the sunlit side of Earth, the sun is first viewed through the atmosphere, and then along an unobstructed path when the spacecraft rises above the horizon. Thus, the measurement sequence during sunrise is just the reverse of that during sunset. In both instances the instrument acquires a measure of attenuation caused by aerosols and gases in the atmosphere, thereby making it possible to quantify these species as a function of altitude.

Typically the spacecraft orbits the Earth approximately once every 90 to 105 minutes or sixteen to fourteen times per day, depending on orbital parameters. Each orbit provides two measurement opportunities, one for each sunset and one for each sunrise. Therefore the instrument can acquire 32 to 28 separate measurements during each 24-hour period, and measurement occurs at different geographical locations over the Earth depending on the spacecraft orbit. The number of measurement opportunities and the geographical coverage can be increased when measurements are made during both lunar and solar occultation events.

Among the 68 satellite instruments studied and analyzed 6 instruments (ILAS, ILAS II, HALOE, SUSIM, SAGE II and SAGE III) make solar occultation measurements. These satellites have a circular or near – circular orbit.

4.5.4 Day / Night Measurements

20 of the satellite instruments make observations only on the day-lit die of the Earth. These instruments generally measure various parameters in the atmosphere by sensing the solar radiation from the atmospheric gases, particles and reflectance from the Earth's surface.

All sunrise-sunset measurement instruments gather data only on the day-lit part of the Earth as they measure vertical profiles of atmospheric optical depth profiles from Earth orbit using the sun as a light source.

The Profiling A-Band Spectrometer/Visible Imager (PABSI) instrument aboard Cloudsat satellite measures the atmospheric radiance of the O_2 A-band rotational spectrum between 761.61 nm and 772.20 nm and records narrow-band images at 747.5 and 761.5 nm. The high-resolution spectrometer determines optical depth and altitude of thin clouds and aerosols by making high spectral resolution (0.5 cm-1) measurements at the Oxygen A-band. A "thicket" of closely spaced spectral lines characterizes the Oxygen A-band. Therefore, a small change in wavelength will vary the rate at which light is attenuated as it traverses the atmosphere. Both the imager and spectrometer are sensitive to reflected sunlight and thus only generate science data on the day side of the Earth.

Another example of a Earth day side only measuring instrumen t is Measurements of Pollution in the Troposphere (MOPITT). MOPITT is the first satellite sensor to use gas correlation spectroscopy. The sensor measures emitted and reflected radiance from the Earth in three spectral bands. As this light enters the sensor, it passes along two different paths through onboard containers of carbon monoxide and methane. The different paths absorb different amounts of energy, leading to small differences in the resulting signals that correlate with the presence of these gases in the atmosphere.

The following chapter gives a detailed design of the satellite instrument data models based on the above measurement modes of the satellite instruments. For an expansion of the satellites and their instruments refer to the Glossary section.

Chapter 5

5 Satellite Instrument Data Model Design

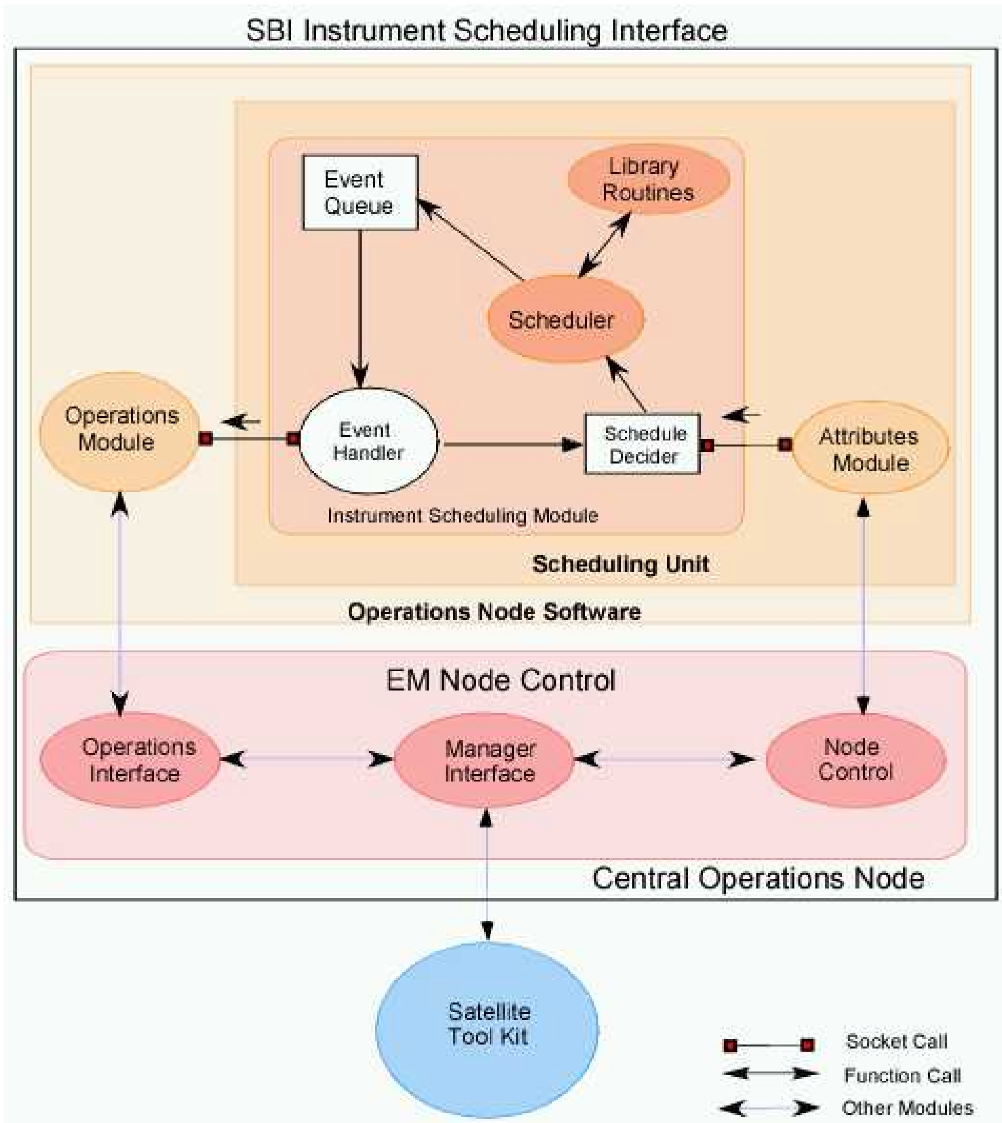
This chapter describes the design of the data model on the Operations Node. Operations Node Software resides on a Central Operations SBI node. The Instrument Scheduling Module interface s of the Operations Node Software shall provide instrument-scheduling information based on the satellite orbital parameters.

There are two interfaces for the Instrument Scheduling Module in the Operations Node Software. The Schedule Decider, within the In strument Scheduling Module, interfaces with the Attributes Module. The Attributes Module is a part of the Scheduling Unit. The Event Handler interfaces with the Operations Module. The chapter also gives details on the file format and subroutine interface to determining Earth Surface types by latitude and longitude.

5.1 Operations Node Instrument Scheduler interface[18]

This interface interfaces with the Attributes Module of the Operations Node to obtain orbital parameters for sche duling the instrument and interfaces with the Operations Module of the Operations Node through which the scheduling commands are passed to the Emulation Manager Node Control Module.

Figure 6. SBI Instrument Scheduling Interface



5.1.1 Instrument Scheduler Application

The Instrument Scheduling Module is an application program that runs on the Operations Node host.

5.1.1.1 Configuration File

All Operation Node applications use the same configuration file to determine information need for communication among the applications. The `OpNodeIsntSched` application retrieves TCP IP socket port numbers from the configuration file in order to connect to the Operations Module and the Node Attributes Module.

5.1.2 Interfaces

There are two interfaces:

- Attributes Module Interface
- Operations Module Interface

5.1.2.1 Attributes Module Interface

The Attributes is a part of the Operations Node Software. It stores the satellite orbital parameters obtained from the Satellite Tool Kit by the EM Node Control Module. The Instrument Scheduling Module obtains the orbital parameters from the Attributes Module to schedule satellite node instruments.

5.1.2.2 Operations Module Interface

The Operations Module is a part of the Operations Node Software. It receives satellite instrument scheduling commands from the Instrument Scheduling Module to be sent to the respective SBI nodes for activating the instruments for data collection and transmission.

Operations Module Message Field Format

The command to the Operations Module has a 'msg' field that specified by the command originator. In this case the command originator is the Instrument Scheduler Module.

5.2 Design Overview

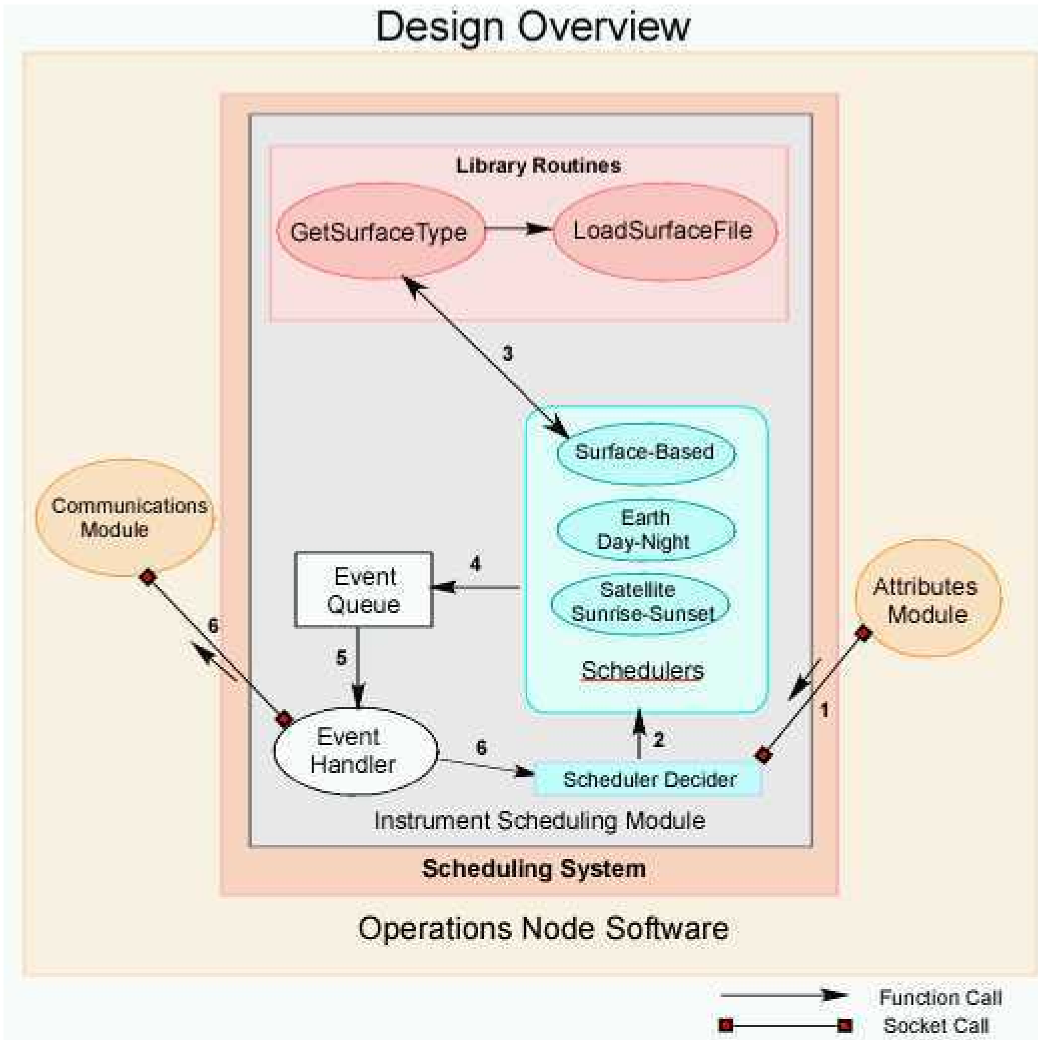
This section contains the design elements of the SBI Operations Node Instrument Scheduling mechanism. The first subsection gives and overview of the design and how it fits into the overall design of the SBI system. The following subsections describe the design of the various types of instrument schedules.

Most satellite instruments gather data based on the Earth surface above, which the satellite passes. Some gather data only over land, some only over water and some over ice (Cryosphere) or a combination of the three.

Therefore the SBI Instrument Scheduling Interface shall be used for scheduling the satellite instruments. This shall be accomplished by interfacing with the Attributes Module, which receives all the orbital parameters from the EM about the nodes in the SBI scenario. Using the orbital parameters, the Earth surface type shall be obtained by calling the library routines.

The surface information shall then be used to decide on the events for scheduling of satellite instruments for data collection. The events shall be stored in the Event Queue from which it shall be passed over to the Communications Module, which then communicates with the Emulation Manager over the Operations Interface.

Figure 7. Scheduler Design – Overview



The Instrument Scheduling Module consists of 3 Schedulers, an Event Queue and Library Routines. The Module gets the orbital parameters, processes the parameters to obtain the time to turn ON or turn OFF the instrument as the case may be and shall put the decision in the Event Queue. The Communications Module shall then process it from the queue. The Instrument Scheduling Module is a part of the Node Program. The N ode Program, Attributes Module and Communications Module are a part of the Operations Node Software.

5.3 Attributes Module

Attributes Module is a part of the Operations Node Software. It shall maintain a table of the satellite orbital parameters for each of t he satellite nodes. Attributes Module interfaces with the Manager interface of the Node control Unit. This module receives all the orbital parameters from the EM about the nodes in the SBI scenario. The SBI Instrument Scheduling Interface interfaces with i t to obtain the orbital parameters.

The orbital parameters with respect to this design that shall be maintained in the node table by the Attributes Module shall be:

- Satellite orbital co-ordinates
- Sunlight times
- Surface value at which the Instrument shall turn ON
- Satellite Orbital Time
- Satellite radius
- Instrument data rate

5.4 Communications Module

Communications Module is the medium for communication between the Operations Node and the other SBI nodes. It passes scheduling information to the Emulation Manager through the operations interface over the dedicated SBI Management Network

5.5 Event Queue

Commands for instrument scheduling generated by the Instrument -Scheduling Module shall be put in this queue. The events are then taken from the queue by the Communications Module, which then passes it to the Operations Interface in the EM Node Control Module.

5.6 Instrument Scheduling module

The Instrument Scheduling module obtains satellite instrument parameters from Attributes Module to calculate the ON or OFF time of t he instrument. The calculated time will be put in an Event Queue. Communications Module then processes the events in the queue.

5.6.1 Operations Node Instrument Scheduler (OpNodeInstSched)[18]

It is a part of the Instrument Scheduli ng module. It interfaces with the Attributes Module to obtain the orbital parameters for scheduling the satellite instruments. It communicates with the Attributes Module by means of a socket connection from the Attributes Module.

The Schedule Decider is called by the Attributes Module using a socket call. It schedules the instruments for the satellite using the SatID passed to it by the Attributed Module. It also receives a SchedID along with the orbital parameters for every satellite instrument. Based on the SchedID it shall pass the parameters obtained from the Attributes Module to the appropriate scheduler.

5.6.1.1 SchedID

SchedID is an integer, which is used to identify a satellite instrument with its scheduler. The SchedID value can be any integer from 0 to 10 and the values are defined as:

0 – Surface based scheduling for instruments which take measurements over only one surface of the Earth.
Example: Vegetation Canopy Lidar (VCL) measures only over the Vegetative land regions

1 – Day night, surface based scheduling for instruments, which take measurements over only one surface of the Earth.
Example: NASA Scaterrometer (NSCAT) does only daytime measurements of the Ocean.

2 – Sunrise sunset scheduling.
Example: Stratospheric Aerosol and Gas Experiment II (SAGE II)

3 – Surface based scheduling for instruments taking measurements over all regions of the Earth.
Example: Cloud Profiling Radar (CPR) of CloudSat satellite

4 – Day night, surface based scheduling for all instruments that take measurements over all surfaces of the Earth only during the day.
Example: Ozone Monitoring Instrument (OMI) of EOS Chem satellite

5 – Surface based scheduling for all instruments that measure over Land and Ocean only
Example: Precipitation Radar (PR) of Tropical Rainfall Measuring Mission (TRMM) satellite

6 – Surface based scheduling for all instruments that measure over Land and Ice only
Example: Visible and near-infrared region (VNIR) of Global Imager (GLI)

7 – Surface based scheduling for all instruments that measure over Ocean and Ice only.
No EOS instruments belong to this type. This is reserved for future use or for satellite instruments of an arbitrary imaginary satellite

8 – Surface based scheduling for all instruments, which measure over vegetative and arid land regions only

Example: Advanced Land Imager (ALI) instrument of Earth Observing 1 (EO 1) does measurements only over vegetative and arid land

9 – Day night, surface based scheduling for all instruments that take measurements over land and ocean only during day time.

Example: Polarization and Directionality of the Earth's Reflectances (POLDER) of Advanced Earth Observing Satellite II (ADEOS II) satellite

10 – Surface based scheduling for all instruments, which measure over vegetative and arid land regions only in the daytime

Example: No EOS instruments belong to this type. This is reserved for future use or for satellite instruments of an arbitrary imaginary satellite

.

5.6.2 Data Scheduling

Scheduling of satellite instruments shall be based on the measurement region of the Earth, by that instrument, which may be of one or a combination of the following:

- Earth surface measurements based on the Earth surface type below
- Satellite sunrise – sunset measurements
- Day / night measurements

Table 6. Instrument Scheduler Parameter Requirements

Parameters obtained From Attributes Module	Measurement Type		
	Earth Surface	Sunrise - Sunset	Day / Night
Orbital Path	Yes	Yes	Yes
Orbital Period	Yes	Yes	No
Surface where it turns ON / OFF	Yes	No	Yes
Daylight Times	No	No	Yes

5.6.2.1 Earth Surface-based Scheduling

Instrument shall be turned ON or OFF depending on the topology below. Parameters obtained from the Attributes Module shall be the orbital co-ordinates of the satellite path, orbital period, earth surface value where the instrument shall be turned ON or OFF, and the ON or OFF value for that surface value. This scheduling may be used in combination with Day / Night scheduling.

5.6.2.2 Satellite Sunrise-Sunset Scheduling

Satellite instrument shall be turned ON during the satellite's sunrise or sunset periods and shall be turned OFF at all other periods. Parameters obtained from the Attributes Module shall be the orbital co-ordinates of the satellite, simulation time and sunlight start & stop times of the satellite. It can be used in combination with Day / Night scheduling.

5.6.2.3 Day / Night Scheduling

Instrument shall be turned ON during Earth day. Instrument shall be turned OFF during Earth night. This scheduling may be used in combination with either of the above mentioned scheduling types.

5.6.3 Program Execution:

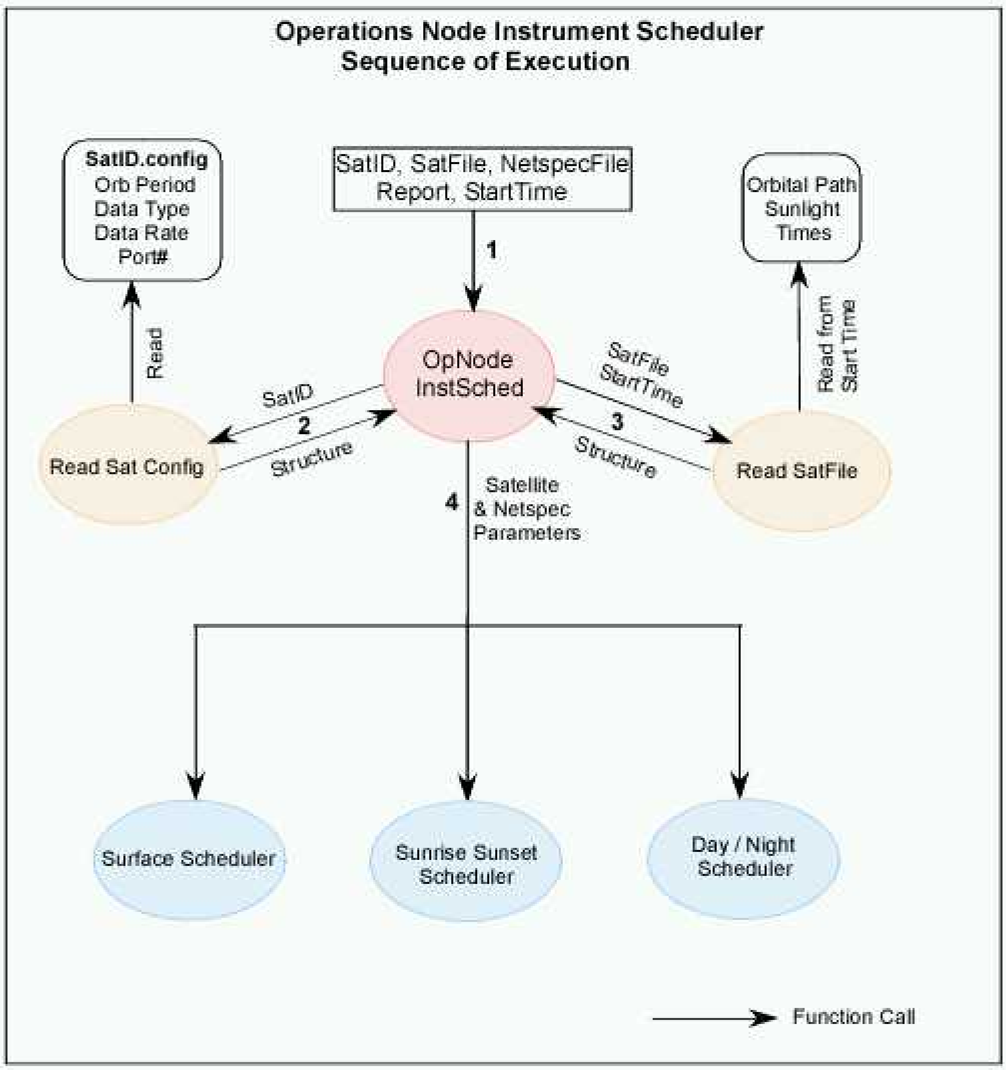
1. The Operations Node Instrument Scheduler OpNodeInstSched is called with the Satellite ID and the satellite configuration files as the argument.

2. OpNodeInstSched calls the ReadSatConfig() routine by passing the SatID as an argument. Satellite configuration parameters are read by ReadSatConfig and stored in a structure.

3. OpNodeInstSched then calls ReadSatFile() with the arguments being the satellite file SchedID and start time of the simulation for that satellite.

Based on the SchedID, for an instrument requiring a Surface-Based scheduler or a Sunrise Sunset scheduler, ReadSatFile() reads the time, latitude, longitude values from the given start time up to one orbit starting from the given latitude and longitude values including the given start time and stores them in a global structure.

Figure 8. Operations Node Instrument Scheduler – Sequence of Execution



In the case of an instrument with SchedID for a satellite instrument measurement based on Day / Night scheduling, ReadSatFile() reads the Sunlight Start and Stop Times, which are then stored into a structure.

4. The program then chooses the appropriate scheduler based on the satellite configuration file that it reads by calling the routine ReadSatConfig().

5.7 Earth Surface – Based Scheduler

Instrument scheduling is done based on the Earth Surface Type for the satellite orbital co-ordinates. The instrument is turned ON over the surface where it is supposed to gather data and turned OFF over the other surfaces of the Earth. For example, the instrument Ocean Color and Temperature Scanner (OCTS) aboard the ADEOS satellite turns ON over the Ocean and is OFF over land and ice (Cryosphere).

The Surface Scheduler is a part of the Instrument Scheduling module. It receives the satellite orbital co-ordinates and orbital period from the Operations Node Instrument Scheduler. Other parameters the Surface Based Scheduler receives from the Operations Node Instrument Scheduler are the Earth Surface value at which the instrument shall be scheduled to turn ON. The instrument is OFF otherwise.

The Earth Surface scheduler obtains surface information (ice, water and land) for the latitude and longitude co-ordinates of the satellite path using the Earth Surface Query Interface.

5.7.1 Earth Surface Query Interface[20]

As mentioned, the Earth Surface Query interface shall provide the surface information (ice, water and land) for the latitude and longitude co-ordinates used for the query. The Central Operations Node shall maintain a database of the Earth's surface. This data is used by satellites that take periodic measurements over ice, water or land to find the next location or co-ordinate where the instrument shall gather data.

The following routines and files are used for this interface. *GetSurfaceType()* is a routine that takes the latitude, longitude and a pointer "surfaceType" passed to it as arguments, retrieves the surface value using the latitude and longitude and stores it in the pointer. The routine *LoadSurfaceFile()* takes the database filename passed to it by the calling subroutine and stores the surface values for every 0.5 degrees latitude and longitude co-ordinates in an integer array. *SurfaceTypeData* is the file in which the surface values are stored. *SurfaceType.h* is the include file which contains the signature of the above two functions and is to be included in the program that calls the programs.

5.7.2 Obtaining a Global Topology Database

A global topology database was needed which could be used to find the surface above which the satellite is, given the satellite's latitude and longitude. This was obtained by processing a World Topology Map of type JPEG using JPEG libraries in C. The pixel color (RGB) values were interpreted into the following topologies using symbols for representing each topology:

Vegetative Land Regions were represented by the "*" symbol and the Arid regions by "&". Ocean was represented with a "." and Cryosphere / Ice represented by a "/"

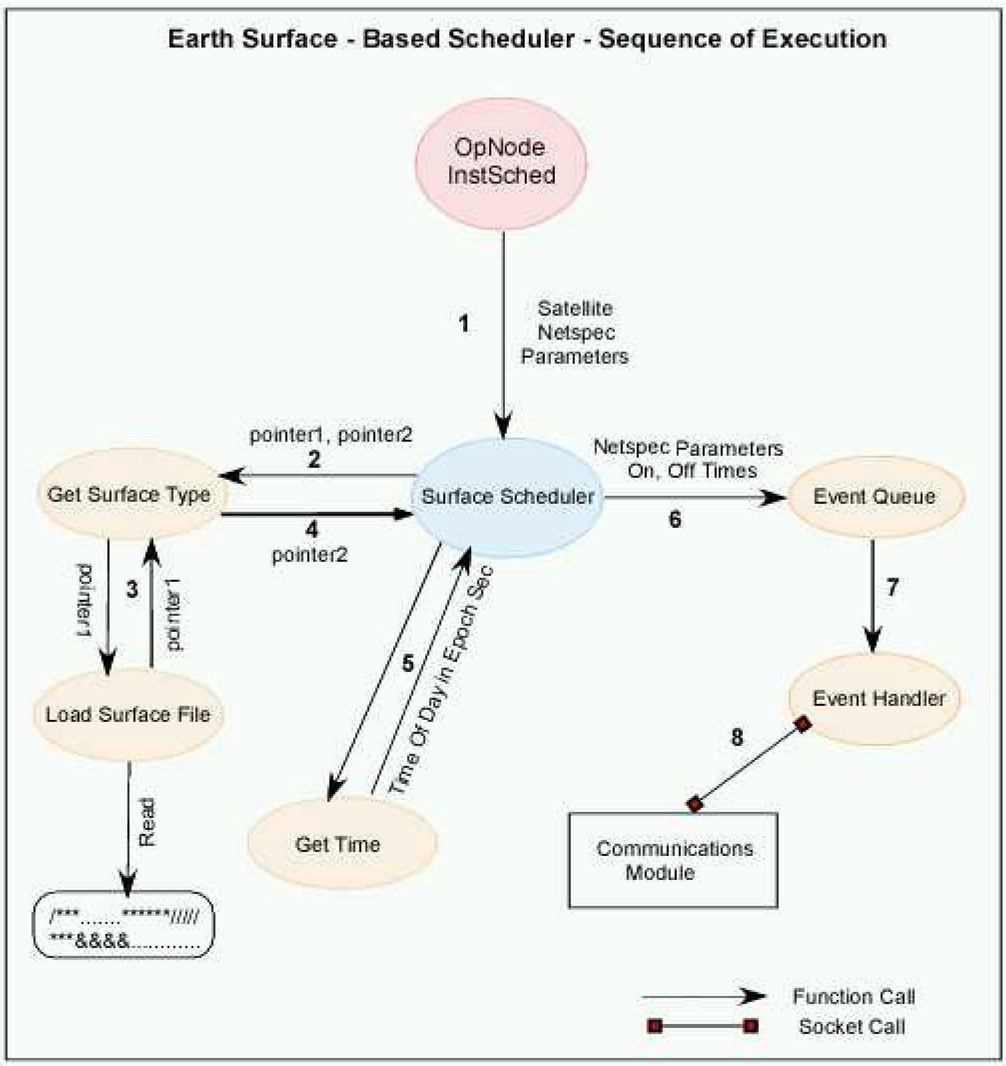
The image file was 721 * 361 pixels wide. The pixel values of the rectangular map were mapped to the latitude and longitude co -ordinates of the Earth. This was stored into a database file: *SurfaceTypeData*

5.7.3 Sequence of Execution

The figure below shows the sequence of execution.

1. The Surface Scheduler is called by OpNodeInstSched based on the SchedID

Figure 9. Earth Surface – Based Scheduler – Sequence of Execution



2. SurfaceScheduler() cal ls routine GetSurfaceType() which obtains the Earth Surface Type for the satellite orbital position

3. GetSurfacetype() calls sub routine LoadSurfaceFile() which loads the Earth surface file: SurfaceTypeData and returns to GetSurfaceType().

4. GetSurfacetype() then obtains the surface values for all the latitude and longitude pairs and stores them in *surfaceType* and returns to SurfaceScheduler()

5. SurfaceScheduler() obtains the current time in Epoch seconds by calling Gettime() subroutine.

6. SurfaceScheduler() ca lculates the On time and OFF time for the satellite instrument and puts it into the Event Queue along with Netspec parameters.

7. Event Handler obtains the parameters from the Event Queue at every ON and OFF time in the Event Queue.

8. Event Handler sends the parameters obtained from the Event Queue to the Communications module via a TCP socket connection.

5.7.4 Earth Surface-Based Scheduler – Algorithm

- Input arguments to the scheduler program are ip addresses of the source and destination, latitude, longitude, time values for one entire orbit (from the starting co-ordinate), the surface value (Act On Surface) at which the instrument is to be turned ON and the satellite orbital period.
- The program obtains the Earth surface values for the orbit co-ordinates it has.
- It then keeps checking the Earth surface values for the successive co-ordinates.
- A counter is incremented until an Earth surface value matches the Act On Surface value, which is obtained as input.
- Knowing the orbital period and the number of co-ordinates that make up a single orbit, the time to get from one co-ordinate to another can be calculated. Thus the time to turn the satellite instrument ON can be calculated using the value of the counter.
- Another counter is incremented from zero until a mismatch with the Act On Surface value is found. This counter is used to calculate the duration for the instrument to be ON.
- Current time (Simulation time) in Epoch seconds is obtained using a sub routine call.
- The ON duration and the OFF duration are added to the current Epoch time to obtain the ON and OFF times to be put into the Event Queue.

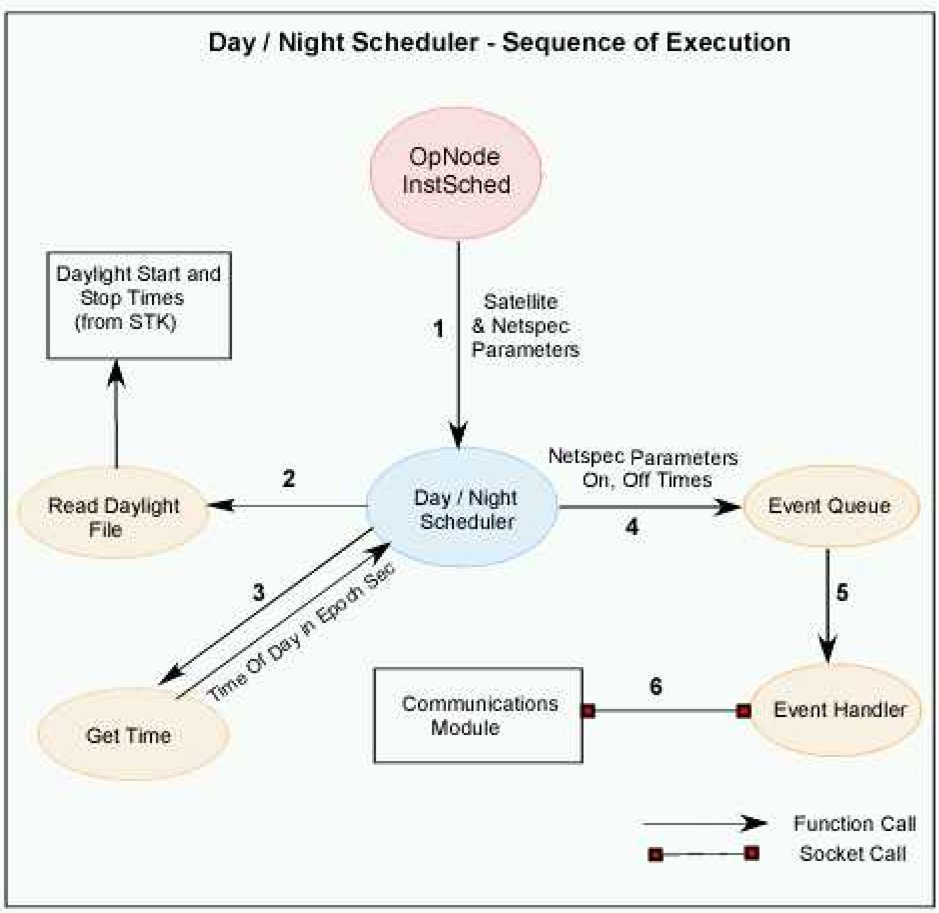
5.8 Day / Night Scheduler

Some satellite instruments are capable of gathering data only when above the sunlit portion of the Earth surface or its atmosphere. The day / night scheduler is used to schedule these instruments. An example would be the Ocean Color Temperature Scanner (OCTS) instrument aboard the ADEOS satellite.

5.8.1 Sequence of Execution

1. The program is started on being called by OpNodeInstSched based on the SchedID. The Sunlight Start and Sunlight Stop times (in Epoch seconds) are passed to the Day / Night Scheduler along with Netspec parameters.

Figure 10. Day / Night Scheduler – Sequence of Execution



2. The scheduler on being started calls the routine ReadDa ylightFile, which obtains the Daylight Start and Stop times.

3. The scheduler then obtains the current simulation time in Epoch seconds by calling the routine GetTime().

4. The Day / Night scheduler uses the times obtained to calculate the duration the instrument will be ON. The ON and OFF times are then calculated, similar to the Earth Surface -based scheduling algorithm. The scheduler ensures that the instrument is scheduled to go ON only after the daylight start time and that it turns OFF before or at daylight stop time.

   For example if the current time is 5:00 PM and the daylight start time is 5:30 PM and stop time is 6:30 PM, then:

   - If ON Time is 5:15 PM, and OFF time is 5:45 PM, ON time becomes 5:30 PM and OFF time remains unchanged
   - If ON Time is 5:15 PM, and OFF time is 6:40 PM, ON time becomes 5:30 PM and OFF time becomes 6:30 PM
   - If ON Time is 5:00 PM and OFF time is 5:20 PM, then the instrument is not scheduled until 6:30 PM

   The instrument is scheduled in steps of one pair of Daylight Start and Stop times. The ON time, OFF time, duration and Netspec parameters are put into the Event Queue.

5. Event Handler obtains the parameters from the Event Queue at every ON and OFF time in the Event Queue.

6. Event Handler sends the parameters obtained from the Event Queu e to the Communications module via a TCP socket connection.

5.8.2 Day / Night Scheduler – Algorithm

- Netspec parameters and the Satellite File name are the input values to the program
- The satellite daylight start and stop times are obtained by a sub routine call.
- Current Epoch time (simulation time) is obtained using sub routine call.
- Surface based scheduling is done and the ON and OFF times obtained from the satellite file and the surface -based scheduling are compared and the appropriate ON and OFF times are determined.
- Duration of ON time is calculated
- The ON, OFF times and the duration are then put on to the Event Queue along with other Netspec parameters obtained from the Schedule Decider (OpNodeInstSched)

5.9 Sunrise Sunset Scheduler

Sunrise Sunset scheduler nee ds to schedule or turn ON the satellite instrument for the satellite sunrise and sunset duration, during which time the instruments measures the sunlight coming through the Earth's atmosphere.

All Sunrise Sunset measurement instruments are turned ON only in Daylight / Sunlight times. They remain OFF during absence of sunlight and if it is not a satellite sunrise or sunset.

It is essential to know the height of the Earth's atmosphere for the calculation of the sunrise and sunset duration of the satellite.

5.9.1 Earth's Atmosphere[21]

The Earth is surrounded by a blanket of air, which we call the atmosphere. It reaches over 560 kilometers (348 miles) from the surface of the Earth, so we are only able to see what occurs fairly close to the ground. Early attempts at studying the nature of the atmosphere used clues from the weather, the beautiful multi-colored sunsets and sunrises, and the twinkling of stars. With the use of sensitive instruments from space, we are able to get a better view of the functioning of our atmosphere.

The atmosphere, solar energy, and our planet's magnetic fields support life on Earth. The atmosphere absorbs the energy from the Sun, recycles water and other chemicals, and works with the electrical and magnetic forces to provide a moderate climate. The atmosphere also protects us from high-energy radiation and the frigid vacuum of space.

The envelope of gas surrounding the Earth changes from the ground up. Four distinct layers have been identified using thermal characteristics (temperature changes), chemical composition, movement, and density.

5.9.1.1 Troposphere

The troposphere starts at the Earth's surface and extends 8 to 14.5 kilometers high (5 to 9 miles). This part of the atmosphere is the most dense. As you climb higher in this layer, the temperature drops from about 17 to -52 degrees Celsius. Almost all weather is in this region. The tropopause separates the troposphere from the next layer. The tropopause and the troposphere are known as the *lower atmosphere*.

5.9.1.2 Stratosphere

The stratosphere starts just above the troposphere and extends to 50 kilometers (31 miles) high. Compared to the troposphere, this part of the atmosphere is dry and less dense. The temperature in this region increases gradually to -3 degrees Celsius, due to the absorption of ultraviolet radiation. The ozone layer, which absorbs and scatters the solar ultraviolet radiation, is in this layer. Ninety-nine percent of "air" is located in the troposphere and stratosphere. The stratopause separates the stratosphere from the next layer.

5.9.1.3 Mesosphere

The mesosphere starts just above the stratosphere and extends to 85 kilometers (53 miles) high. In this region, the temperatures again fall as low as -93 degrees Celsius as you increase in altitude. The chemicals are in an excited state, as they absorb energy from the Sun. The mesopause separates the stratosphere from the next layer. The regions of the stratosphere and the mesosphere, along with the stratopause and mesopause, are called the *middle atmosphere* by scientists. This area has been closely studied on the ATLAS Spacelab mission series.

5.9.1.4 Thermosphere

The thermosphere starts just above the mesosphere and extends to 600 kilometers (372 miles) high. The temperatures go up as you increase in altitude due to the Sun's energy. Temperatures in this region can go as high as 1,727 degrees Celsius. Chemical reactions occur much faster here than on the surface of the Earth. This layer is known as the *upper atmosphere*.

The upper and lower layers of the thermosphere will be studied more closely during the Tethered Satellite Mission (TSS-1R). As the EOS satellites study the gases and their properties, the height of the Earth’s atmosphere is around 85 km up to the edge of the Mesosphere.

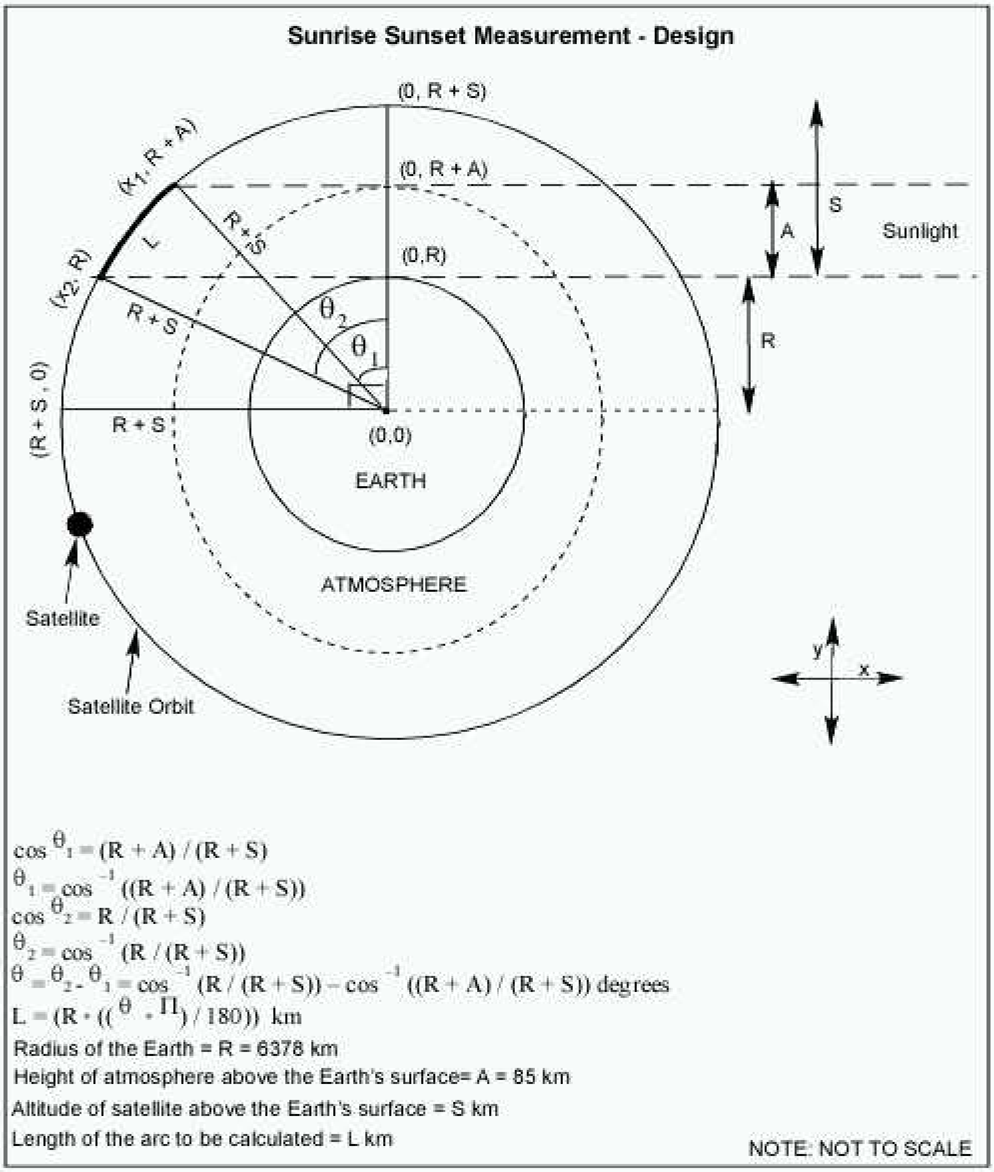
5.9.2 Design Analysis

The satellites with the sunrise-sunset measurement instruments have a circular or near circular orbit. Due to the large distance between the earth and the sun it can be assumed that the sunlight that the sunray s hitting the Earth's atmosphere are parallel to each other. The design below is based on this assumption.

The following are the variables or notations used in the design:
R is the Equatorial radius of the Earth (6378 km)
A is the height of the Earth's atmosphere above the surface of the Earth (85 km)
S is the height of the satellite above the surface of the Earth (obtained from STK)

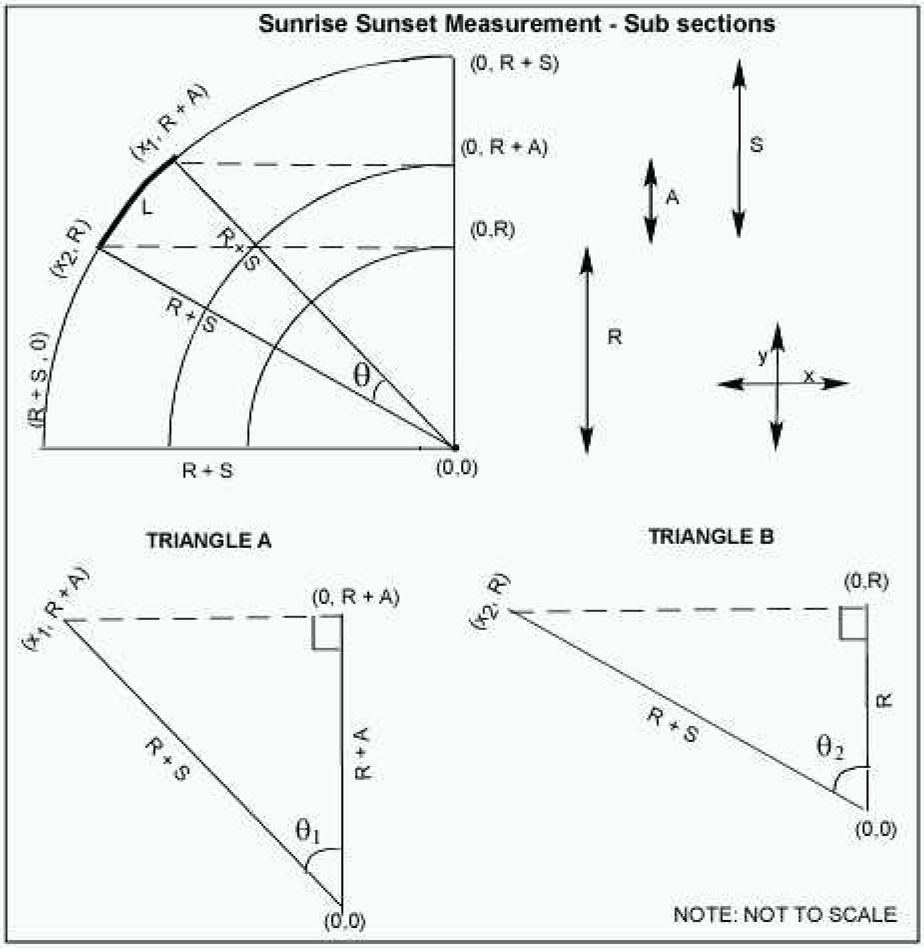
The length of the dark arc of the satellite's orbit in the figure below is required to calculate the sunrise duration.

Figure 11. Sunrise Sunset Measurement – Design



The diagram below gives a dissection of the diagram above in order to have a clear view of the design.

Figure 12. Sunrise Sunset Measurement – Sub sections



In the diagram above, TRIANGLE A shows that θ_1 can be obtained using the fundamentals of trigonometry.

$$\cos \theta_1 = (R + A) / (R + S)$$

$$\theta_1 = \cos^{-1} ((R + A) / (R + S))$$

Similarly in TRIANLGE B θ_2 can also be obtained

$$\cos \theta_2 = R / (R + S)$$

$$\theta_2 = \cos^{-1} (R / (R + S))$$

θ can be obtained by simply subtracting θ_1 from θ_2.

$$\theta = \theta_2 - \theta_1 = \cos^{-1} (R / (R + S)) - \cos^{-1} ((R + A) / (R + S)) \text{ degrees}$$

If θ is known then the length L of the arc of the circle may be determined:

$$L = (R * \theta) * (\pi / 180) \text{ km}$$

The circumference of the satellite orbit is determined

$$C = 2 * \pi * (R+S)$$

Where,

C is the Circumference of the satellite orbit

Knowing orbital period (O) of the satellite in seconds, the time to cover the arc is calculated.

$$T = (L * O) / C \text{ seconds}$$

Where,

T is the time taken in seconds by the satellite to cover the distance L km

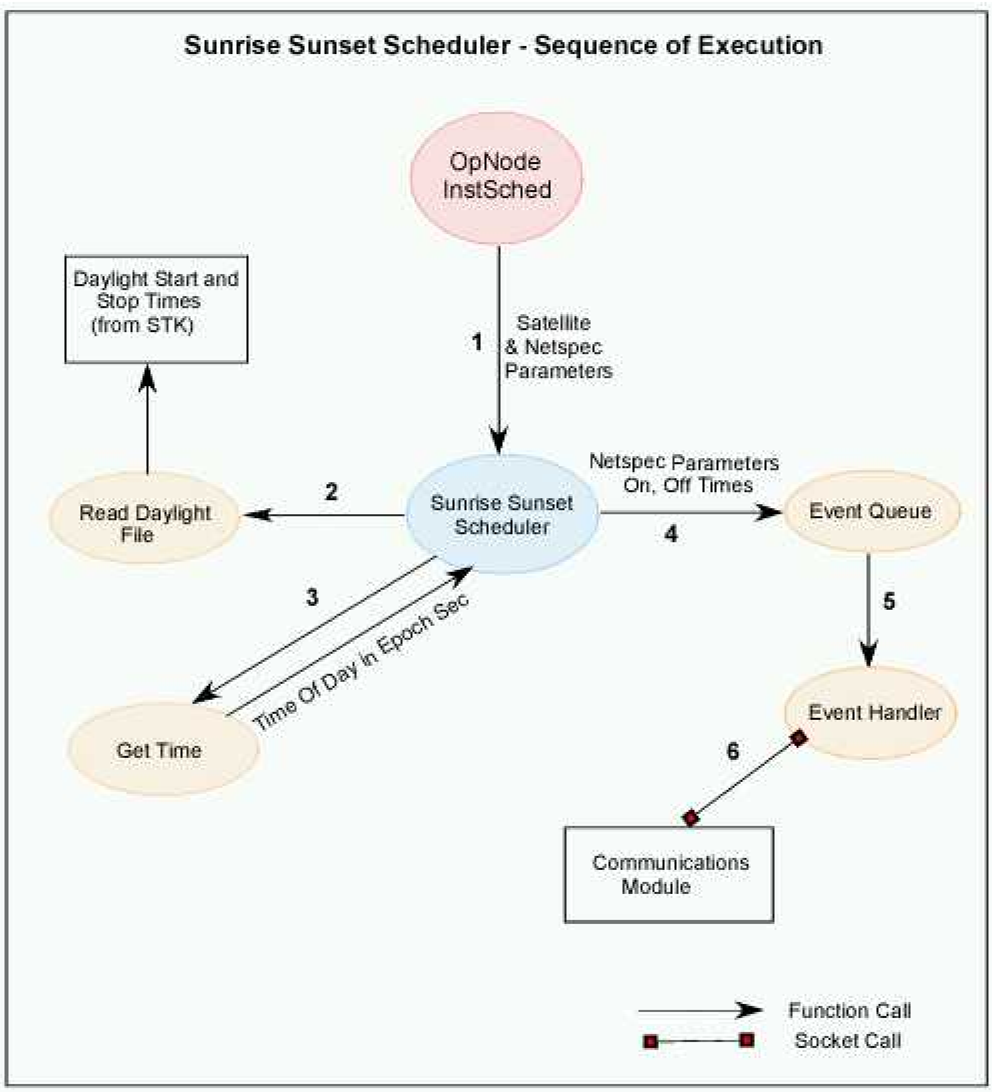
For the satellite sunrise, Sunlight start time (ON time) is added to T to obtain the OFF time in the case of the satellite sunrise. In the case of a satellite sunset, the time T is subtracted from the Sunlight stop time (OFF time) to obtain the instrument ON time.

5.9.3 Sequence of Execution

The program is called by the Operation Node Instrument Scheduler based on the SchedID of the instrument.

1. The scheduler on being started calls the routine ReadDaylightFi   le() and obtains the Daylight Start Time and Daylight Stop time

Figure 13. Sunrise Sunset Scheduler – Sequence of Execution



2. The scheduler then obtains the current simulation time in Epoch seconds by calling the routine GetTime().

   Using the current simula tion time and the Sunlight Start Time it calculates the time left to turn the instrument ON. The OFF time is calculated as explained in section 1.9.2. Similarly, using the current simulation time and the Sunlight Stop Time it calculates the time left to tu rn the instrument OFF. ON time is calculated as explained in section 1.9.2

3. The above determined ON, OFF values and the duration for the instrument to be ON is then put into the Event Queue along with the Netspec parameters.

4. Event Handler obtains the para meters from the Event Queue whenever an ON or OFF event is to occur.

5. The handler passes the parameters to the Communications module via a TCP socket connection.

Chapter 6

6 SBI Node Scheduler Design

This chapter describes the design of the Instrument Scheduler on the SBI Nodes after scheduling has been done by the Operations Node. It is a part of the Common Node Software, which resides on all the SBI nodes. The Instrument Scheduling Module interfaces of the SBI Node Software shall generate traffic based on the parameters obtained from the Central Operations Node.

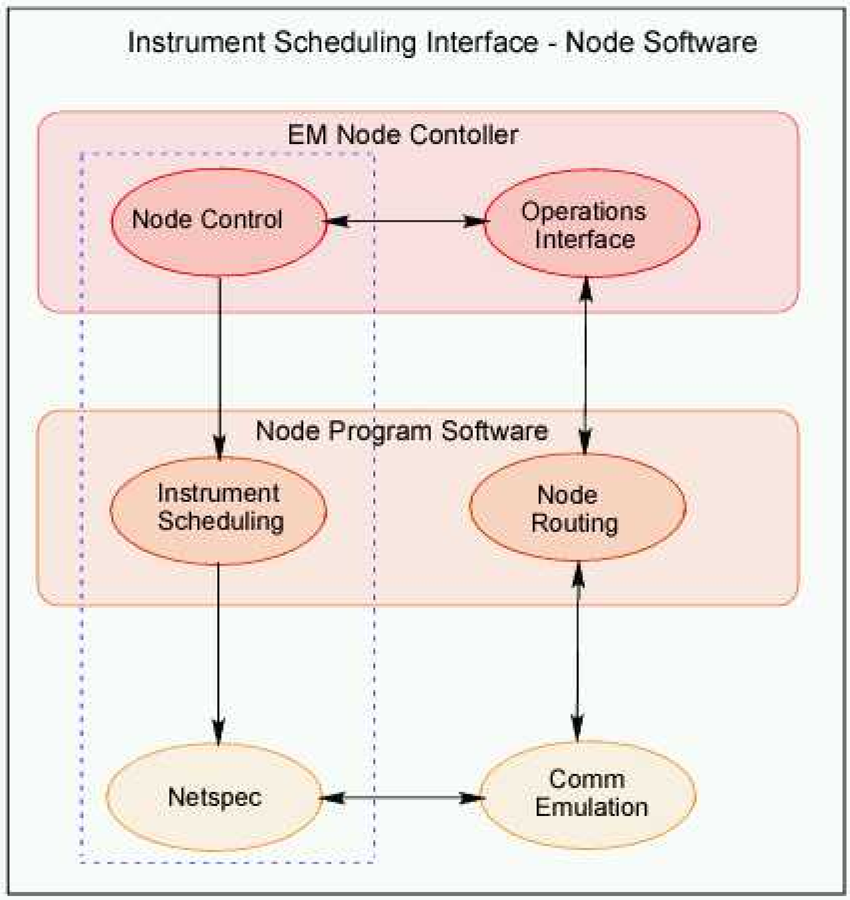
6.1 SBI Node Instrument Scheduling Interface[19]

The SBI Node Instrument Scheduling Interface is a single subroutine call. This routine is called from a satellite 'node' application program running on each SBI Satellite node. The Instrument Scheduling module of the Node Program Software schedules the satellite's instruments based on the information obtained from the Operations Node Instrument Scheduling Module software via the Operations Channel.

The specifications reflect a rapid prototype with significant 'Netspec' specification data in the Operation Node to Satellite Node Instrument command. This command can be modified in the future to come closer to an instrument specific command (without Netspec specific data in it). The Instrument Scheduling Module interfaces with the network data-modeling program 'Netspec' for generation of emulated instrument data network traffic.

6.2 Overview of the SBI Scheduling Interface

Figure 14. Instrument Scheduling Interface – Node Software



The diagram above shows the Instrument scheduling module and its interfaces in the dotted box.

6.2.1 Node Control Module

The Node Control Module receives the start, stop, pause and r esume from the manager interface of the Node control Unit. It receives scheduling information from the Operations Node through the Operations Interface

6.2.2 Instrument Scheduling Module

The instrument scheduling module receives scheduling information from the Node Control Module. It writes the Netspec file based on the scheduling information obtained from Node Control Module. It sends the Netspec file to the Netspec Module.

6.2.3 Netspec Module

Netspec shall be used to simulate traffic flow between satellites or satellites and ground stations. Netspec Module receives the Netspec file from the Instrument Scheduling Module. It shall run Netspec locally on the source node. Similarly Netspec shall be run locally on the destination node.

6.2.4 Interface between SBI Node Instrument Scheduler and Node Control Modules

This interface shall be a function call from the Node Control Module. Instrument Scheduling Module receives the parameters required to write the Netspec script and writes it onto a file. It shall then send the file to the Netspec Module

The Satellite Node Instrument Scheduler Module is a dynamically link library loaded by the 'node' application. The library name, library include file, and subroutine interface are described in this section.

The shared library of the Satellite Node Instrument Scheduling Module has the name 'libNodeInstSched.so' and resides in the 'lib' directory of the SBI software directory hierarchy.

The C include file for defining the interface to the Satellite Node Instrument Scheduler Module library is named 'NodeInstSched.h' and resides in the 'include' directory of the SBI software directory hierarchy.

6.2.5 Interface between SBI Node Instrument Scheduler and Netspec Module

The interface shall be a function call NetspecWrapper from the Instrument Scheduling Module. The Instrument Scheduling Module shall call Netspec with the Netspec file as an argument. The Netspec Module shall run Netspec using that file.

6.3 Design of SBI Node Scheduler

The Satellite node instrument scheduler consists of the routine *NodeInstSched* that shall receive the data through a socket from the Node Control Module. The program arguments are specified above in section 1.2.4.1

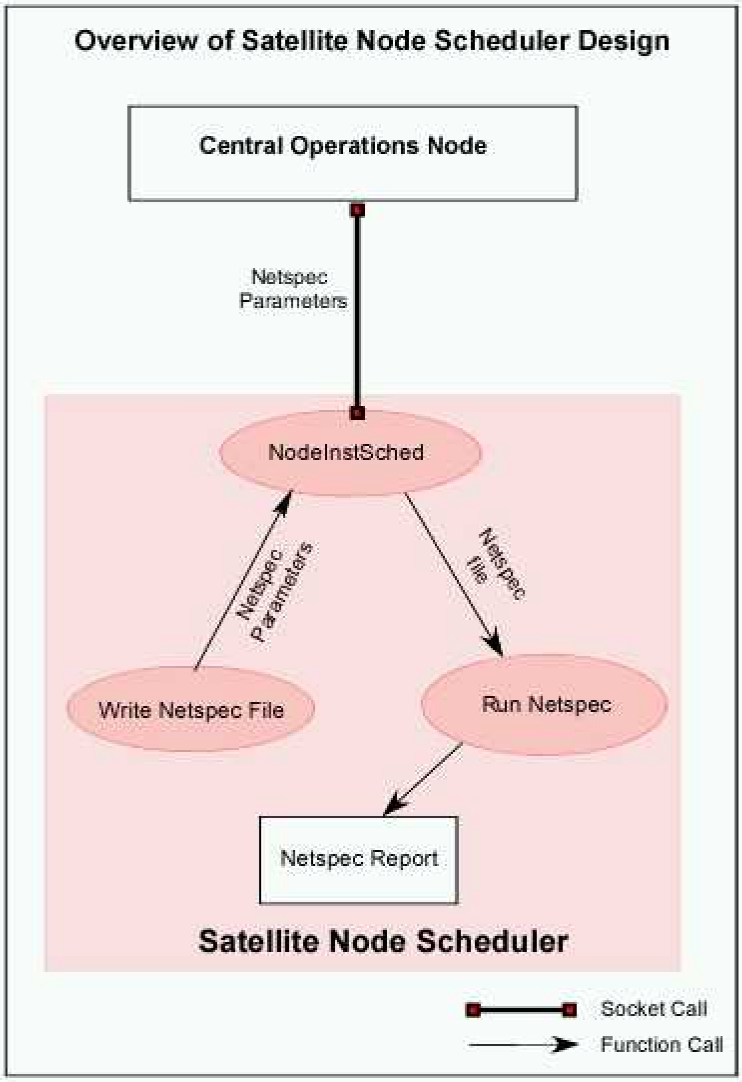
The *NodeInstSched* shall check the Message / Data type and shall call the appropriate subroutine to write ou t a Netspec file. On successful completion of the Netspec script, the *NodeInstSched* shall call the *RunNetspec* subroutine to run Netspec

6.3.1 Overview

The design diagram below depicts the data flow. The interface between the Operation Node Scheduler and the Sat ellite Node Scheduler is a socket. The scheduling information for that node shall be received through the socket. Using this information, the Netspec script is written and is then scheduled to run between the source and destination nodes.

Netspec paramete rs are sent to the appropriate SBI Node by the Central Operations Node via a socket interface. The NodeInstSched routine receives the message and passes the Netspec information to WriteNetspecFile. WriteNetspecFile is a routine, which calls the appropriat e Netspec script generator routine based on the DataType value obtained in the message. The script generator routine generates a Netspec script based on the duration and throughput values obtained in the message received from the Operations Node.

Figure 15. Overview of Satellite Node Scheduler Design



The NodeInstSched then calls RunNetspec routine, which executes the Netspec script, generated and logs the Netspec report that is generated. The following section gives a description of the library routines used.

6.4 Library Routines

6.4.1 WriteNetspecFile routine

This routine calls one of 6 different types of subroutines depending on the value of *DataType* received as shown in the table below:

Table 7. Data Type Values

DataType	TrafficType
0	Burst
1	qburst
2	mpeg
3	video teleconferencing
4	voice
5	www

The appropriate routine is called to write out the Netspec file.

Depending on the DataType value the Program Name shall be one of the following:

- `burst_script_gen( )`
- `qburst_script_gen( )`
- `voice_script_gen( )`
- `mpeg_script_gen( )`
- `vidtel_script_gen( )`
- `www_script_gen( )`

6.4.2 Network Communications

Communications Module is the medium for communication between the Operations Node and the other SBI nodes. It passes scheduling information to the Emulation Manager t hrough the operations interface over the dedicated SBI Management Network. Communication is through a socket interface. Netspec data traffic is sent over UDP to the destination satellite / ground station

6.4.3 Implementation Restrictions

Netspec takes only a who le integer for duration and throughput. Therefore, duration and throughput shall be approximated to the nearest whole integer

Chapter 7

7 Test Results & Evaluation

The Instrument schedulers were first tested individually to ensure that the results are as expected. The following sections give some test results run on each of the 3 schedulers separately

7.1 Test Scenario

The individual scheduler tests were done using satellite parameters from the Upper Atmosphere Research Satellite (UARS). The parameters were obtained using the Satellite Tool Kit (STK). Each scheduler has been tested for all possible output results that were expected. Land when referred to in general implies both the arid and vegetative lands. The initial scheduler tests are done by using two nodes namely, bluenode02 as source and bluenode01 as the destination. The tests have been done using both TCP and UDP. Each scheduler schedules one pair of ON and OFF times within a single orbit, starting from the given time.

The UARS configuration file had the following parameters:

SchedID that decides what type of scheduling has to be done which can be varied for running the tests.

Orbital Period of the UARS satellite is 96 minutes, which is 5760 seconds. This is fixed.

Altitude of UARS satellite is 566 km

Surface to act on (*ActOnSurface*) which may be varied according to the test condition

Data Type, which is set initially as burst mode (0), but can be varied according to the instrument.

Data Rate is varied to suit the test conditions. It is in kilobits per second (Kbps)

Port number to be used by Netspec for generating traffic between the two satellites

7.2 Test Constraints

- The link bandwidth between the nodes used in the test is approximately 100 Mbps provided the link is not utilized by other nodes.
- There is a duration limitation 2000 seconds. (2147.48364 to be exact) in Netspec. This is because Netspec maintains time in microseconds and so at this value a signed int overflows. Therefore, any duration that is greater than 2000 seconds is scheduled for 2000 seconds. The event handler shall call the scheduler after 2000 seconds to schedule the instrument again.
- The system configuration had to be changed for tests in UDP mode with required throughput > 60 Mbps. If the throughput value is > 60 Mbps, the burst period in Netspec script is below the jiffy value of a 2.4.6 kernel on Intel platform (10 milliseconds). Therefore a KU Real Time (KURT) kernel was used, which has a jiffy value of about 1 millisecond. This problem was not encountered for the TCP mode.

7.3 System Configuration

7.3.1 Burst and Queued burst traffic types

System Configuration for TCP tests:

RedHat version: 7.1

Kernel Version: 2.4.6

eth2 was created on bluenodes 1 and 2 and veth0 (virtual Ethernet) was created on eth2.

Source: veth0 of bluenode02

Destination: veth0 of bluenode01

Bandwidth: 100 Mbps (approximately)

System Configuration for UDP tests with throughput <= 60 Mbps or burst period >= 10 milliseconds:

RedHat version: 7.1

Kernel Version: 2.4.6

eth2 was created on bluenod es 1 and 2 and veth0 (virtual Ethernet) was created on eth2.

Source: veth0 of bluenode02

Destination: veth0 of bluenode01

Bandwidth: 100 Mbps (approximately)

System Configuration for UDP tests with throughput > 60 Mbps or burst period < 10 milliseconds:

RedHat version: 7.1

Kernel Version: 2.4.13-KURT

Source: bluenode02

Destination: bluenode01

Bandwidth: 100 Mbps (approximately)

System configuration for both TCP and UDP tests for VBR traffic (MPEG and Video Teleconference):

For VBR traffic, Netspec script s were found to be unable to generate any throughput in the bluenodes kernel used for the tests. They were found to work on the following system configuration:

RedHat Version: 6.2

Kernel Version: 2.2.14-5.0.

Source: testnode9

Destination: testnode12

Bandwidth: 100 Mbps (approximately)

7.4 Surface Based Scheduler Tests

Surface scheduler must be capable of handling the following events:

- Scheduling Vegetative Land or Arid Land or Ocean or Ice (SchedID 0)
- Scheduling Land and Ocean (SchedID 5)
- Scheduling Land and Ice (SchedID 6)
- Scheduling Ocean and Ice (SchedID 7)
- Scheduling Arid and Vegetative Land (SchedID 8)
- Scheduling Land, Ocean and Ice (SchedID 3)

NA in the tabulated results below means Not Applicable. No values were generated for that field in the tables. The following test scenarios were done to test the scheduler under the above condition.

7.4.1 Scheduling Vegetative Land or Arid Land or Ocean or Ice

The results have been tabulated and below it is the explanation of the test results. The table shows the start up configuration file values used for the tests done in both TCP and UDP.

Table 8. Surface based scheduler: Test Configuration

#	Scheduler ID	Orbital Period secs	Altitude Km	Surface to turn ON	Data Type	Required Data Rate	Scheduler Onset time secs
1	0	5760	566	0	0	15 Mbps	5000
2	0	5760	566	1	0	15 Mbps	5000
3	0	5760	566	2	0	5 Mbps	5000
4	0	5760	566	3	0	25 Mbps	5000
5	0	5760	566	1	0	50 Mbps	6000
6	0	5760	566	1	0	50 Mbps	7000
7	0	5760	566	2	0	70 Mbps	0
8	0	5760	566	3	0	90 Mbps	1000
9	0	5760	566	2	0	95 Mbps	2000

The scheduler ID 0 means that the scheduler is going to schedule for an instrument that measures over just a single topology of the Earth. Scheduler Onset Time is the simulation time in seconds from which the scheduler is set to schedule one pair of ON and OFF times for the instrument.

Table 9. Surface – Based Scheduler: Results

#	Surface Turned ON	ON Time secs	OFF Time secs
1	0	NA	5760
2	1	0	335.5
3	2	335.53	351.511
4	3	351.512	455.368
5	1	0	1318.17
6	1	0	319.556
7	2	1350.12	1621.75
8	3	623.134	647.101
9	2	0	31.955

The table above gives the ON and OFF times as scheduled by the scheduler. The ON time is the time from scheduler onset time when the instrument is turned ON. Similarly the OFF time is time in seconds from the scheduler on set time when the instrument is to be turned off.

<u>TCP Results:</u>

Table 10. Surface – based scheduler: TCP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	NA	5760	NA	NA	15 Mbps	NA	NA
2	335.53	335.53	335.349	335.35	15 Mbps	14.985 Mbps	14.985 Mbps
3	16.01	16	16.029	16.03	5Mbps	4.994 Mbps	4.994 Mbps
4	103.85	104	104.16	104.16	25 Mbps	24.964 Mbps	24.964 Mbps
5	1318.17	1318	1319.2	1319.2	50 Mbps	49.955 Mbps	49.955 Mbps
6	319.556	319	319.393	319.39	50 Mbps	49.940 Mbps	49.940 Mbps
7	271.623	271	271.305	271.31	70 Mbps	69.924 Mbps	69.924 Mbps
8	23.967	24	24.087	24.089	90 Mbps	89.712 Mbps	89.706 Mbps
9	31.955	31	33.669	33.67	95 Mbps	87.611 Mbps	87.608 Mbps

The Expected Duration is the durati on that is expected to be set by the scheduler. Calculated duration is the duration set / calculated by the scheduler after approximating the duration (Netspec implementation restrictions: see 6.4.4). The actual duration are the duration of transmission of data by the transmitter (TX – bluenode02) and the duration of reception by the receiver (RX – bluenode01).

Expected throughput is the throughput that is required of the instrument. Actual throughput is the transmitted throughput by the transmitter / sour ce and the throughput received by the receiver / destination.

Test conditions:

In test # 1, the starting time and the ActOnSurface were chosen such that, the surface (ice) is not found during one entire orbit of the satellite, from the given scheduler onset time. The expected result would be an output, which gives the time to remain off for the entire orbit, beginning from the Scheduler Onset Time. The scheduler gave an output of 5760 as time to remain OFF, which is the expected result.

The other tests t est the maximum capable traffic generation using Netspec in TCP mode and the duration of the ON time. It can be seen from the tabulated results above, that Netspec can generate up to 90 Mbps traffic throughput.

UDP Results:

Table 11. Surface – based scheduler: UDP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	NA	5760	NA	NA	15 Mbps	NA	NA
2	335.53	335.53	335.33	335.33	15 Mbps	14.975 Mbps	14.975 Mbps
3	16.01	16	16.04	16.04	5Mbps	4.991 Mbps	4.991 Mbps
4	103.85	104	104.151	104.15	25 Mbps	24.966 Mbps	24.966 Mbps
5	1318.17	1318	1319.2	1319.2	50 Mbps	49.955 Mbps	49.955 Mbps
6	319.556	319	319.403	319.41	50 Mbps	49.938 Mbps	49.938 Mbps
7	271.623	271	271.689	272.69	70 Mbps	69.824 Mbps	69.823 Mbps
8	23.967	24	24.111	24.116	90 Mbps	89.605 Mbps	89.587 Mbps
9	31.955	31	31.128	31.134	95 Mbps	94.624 Mbps	94.610 Mbps

It is seen that UDP is capable of generating more traffic. It can be seen from the above results that the scheduling and d ata generation is as expected and is accurate in the case of UDP (particularly for high data rates) and is accurate for TCP up to 90 Mbps.

7.4.2 Scheduling for more than one Earth surface type

This involves scheduling for instruments with SchedID (scheduler ID) as 3, 5, 6, 7 or 8.

3 – Scheduling for Land, Ocean and Ice

5 – Scheduling for Land and Ocean measurements

6 – Scheduling for Land and Ice measurements

7 – Scheduling for Ocean and Ice measurements

8 – Scheduling for Arid and Vegetative Land measurements

Testing was done for each of the above. The following were the start up configuration file values for both TCP and UDP modes.

Table 12. Other Surface – Based scheduler Tests: Configuration

#	Scheduler ID	Orbital Period secs	Altitude km	Surface to turn ON	Data Type	Required Data Rate	Scheduler Onset Time Secs
1	5	5760	566	1,2,3	0	1 Kbps	5660
2	6	5760	566	0,2,3	0	60 Mbps	0
3	7	5760	566	0,1	0	80 Mbps	2600
4	8	5760	566	2,3	0	10 Kbps	2000
5	3	5760	566	0.1.2.3	0	90 Mbps	0

The results were as follows:

Table 13. Other Surface – Based scheduler Tests: Results

#	Surface turned ON	ON Time secs	OFF Time secs
1	1,2,3	5660	7660
2	0,2,3	1350.1	1701.64
3	0,1	2600	2735.81
4	2,3	2000	2031.96
5	0.1.2.3	0	2000

<u>TCP Results:</u>

Table 14. Other Surface – Based scheduler Tests: TCP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	2000	2000	2001.79	2001.75	1 Kbps	1 Kbps	1 Kbps
2	351.512	351	351.334	351.333	60 Mbps	59.945 Mbps	59.945 Mbps
3	135.81	135	135.227	135.227	80 Mbps	79.872 Mbps	79.872 Mbps
4	31.955	31	31.059	31.059	10 Kbps	10 Kbps	10 Kbps
5	2000	2000	2001.98	2001.97	90 Mbps	89.912 Mbps	89.912 Mbps

UDP Results:

Table 15. Other Surface – Based scheduler Tests: UDP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	2000	2000	2001.77	2001.76	1 Kbps	1 Kbps	1 Kbps
2	351.512	351	351.828	351.828	60 Mbps	59.860 Mbps	59.859 Mbps
3	135.81	135	135.364	135.367	80 Mbps	79.791 Mbps	79.789 Mbps
4	31.955	31	31.04	31.039	10 Kbps	10 Kbps	10 Kbps
5	2000	2000	2005.65	2005.64	90 Mbps	89.745 Mbps	89.747 Mbps

In the case of test #1 and 5, the scenario was set the duration of the ON time was actually greater than 2000 seconds. The results show th at the duration has been cut short to 2000 seconds, due to the limitation in Netspec.

7.5 Day Night Scheduler Tests

In the case of Day Night Scheduler, the scheduler calls the `ReadDaylightFile()` sub routine, which reads the daylight start and stop times from the daylight file obtained from STK.

Table 16. Daylight File

Daylight Start Time (EpSec)	Daylight Stop Time (EpSec)
0	1727.11
3862.13	7489.62
9624.5	13252.12
15386.85	19014.63
21149.18	24777.13
26911.5	30539.63
32673.8	36302.13
38436.09	42064.63
44198.36	47827.13
49960.61	53589.63
55722.85	59352.13
61485.07	65114.63
67247.27	70877.14
73009.46	76639.64
78771.63	82402.14
84533.79	86400

The daylight file contains the Daylight Start and Stop Times in Epoch seconds.

For the purpose of testing, the following situations were tested:

- Giving the Scheduler Onset time between the Daylight Start and Stop times
- Giving the Scheduler Onset time between Daylight Stop and Start times
- Scheduler Measurement of any one of the Earth surfaces (SchedID 1)
- Scheduler measurement of both Land and Ocean (SchedID 9)
- Scheduler measurement of both Arid and Vegetative land (SchedID 10)
- Scheduler measurement of all of the Earth Surfaces (SchedID 4)

7.5.1 Scheduling for only one Earth surface type

The tests below investigate the res ults obtained when checking for the first 3 conditions bulleted above.

The table below gives the various fields in the start up configuration file:

Table 17. Day / Night Scheduler Test1: Configuration

#	Scheduler ID	Orbital Period secs	Altitude km	Surface to turn ON	Data Type	Required Data Rate	Scheduler Onset Time secs
1	1	5760	566	0	0	15 Mbps	5000
2	1	5760	566	1	0	25 Mbps	2000
3	1	5760	566	2	0	90 Mbps	0
4	1	5760	566	3	0	50 Mbps	200

Table 18. Day / Night Scheduler Test1: Results

#	Surface turned ON	ON Time secs	OFF Time secs
1	0	NA	5760
2	1	2032	2734.98
3	2	1350.1	1621.74
4	3	1622	1645.99

TCP Results:

Table 19. Day / Night Scheduler Test1: TCP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	NA	5760	NA	NA	15 Mbps	NA	NA
2	NA	NA	NA	NA	25 Mbps	NA	NA
3	271.616	271	271.288	271.292	90 Mbps	89.908 Mbps	89.906 Mbps
4	23.966	23	23.038	23.04	50 Mbps	49.940 Mbps	49.936 Mbps

Test #1: The Scheduler Onset Time is set such that it is between Daylight Start and Stop times, with Earth surface Ice (Act On Surface 0) to turn on the instrument.

The orbit at that time did not go over Ice. The result shows that the time calculated was the time to stay OFF (5760 seconds) which was one orbital period of the satellite, from Scheduler Onset Time.

Test #2: The scenario was set such that the time to go ON was after Daylight Stop Time and OFF was before Daylight Start Time. Therefore, the scheduler should not schedule the ON and OFF times. The scheduler output as seen above, shows no traffic generated. The ON, OFF times shown are for reporting purposes only. The ON time is 2000 seconds and the OFF time is 2734.98 seconds.

Test #3 & #4: The Scheduler Onset Times were set between daylight start and stop times. The results were as expected.

UDP Results:

The UDP results were similar to the TCP results.

Table 20. Day / Night Scheduler Test1: UDP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	NA	5760	NA	NA	15 Mbps	NA	NA
2	NA	NA	NA	NA	25 Mbps	NA	NA
3	271.616	271	271.628	271.685	90 Mbps	89.776 Mbps	89.773 Mbps
4	23.966	23	23.047	23.1	50 Mbps	49.920 Mbps	49.934 Mbps

7.5.2 Scheduling for more than one surface type

Table 21. Day / Night Scheduler Test2: Configuration

#	Scheduler ID	Orbital Period secs	Altitude km	Surface to turn ON	Data Type	Required Data Rate	Scheduler Onset Time secs
1	9	5760	566	1,2,3	0	60 Mbps	5480
2	10	5760	566	2,3	0	100 Kbps	3650
3	4	5760	566	0.1.2.3	0	9 Mbps	1600

The results were as follows:

Table 22. Day / Night Scheduler Test2: Results

#	Surface turned ON	ON Time secs	OFF Time secs
1	1,2,3	5480	7480
2	2,3	5000	5351.64
3	0.1.2.3	1600	1727.1

TCP Results:

Table 23. Day / Night Scheduler Test2: TCP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	2000	2000	2001.81	2001.8	60 Mbps	59.946 Mbps	59.946 Mbps
2	351.637	351	351.319	351.317	100 Kbps	100 Kbps	100 Kbps
3	127.1	127	127.109	127.109	9 Mbps	8.993 Mbps	8.993 Mbps

Test #1: The instrument had to take measurements over both land and ocean. The Scheduler Onset time was between the Daylight Start Times.

Test #2: The instrument takes measurements over Land (Arid and Vegetative regions). The Onset time was before Daylight Start Time.

Test #3: The instrument takes measurements throughout the day. The scheduler was expected to set the Daylight Stop time as OFF time.

The outcome of the results were as expected. The UDP and TCP results show that required / expected traffic was generated.

UDP Results:

Table 24. Day / Night Scheduler Test2: UDP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	2000	2000	2005.66	2005.65	60 Mbps	59.831 Mbps	59.831 Mbps
2	351.637	351	351.46	351.357	100 Kbps	100 Kbps	100 Kbps
3	127.1	127	127.13	127.13	9 Mbps	8.992 Mbps	8.992 Mbps

7.6 Sunrise Sunset scheduler tests

The sunrise sunset scheduler makes use of the Daylight file just like the Day Night scheduler. The following cases may be tested:

- Giving the Scheduler Onset time between the Daylight Start and Stop times
- Giving the Scheduler Onset time between Daylight Stop and Start times

Test conditions:

The surface to turn ON will not be needed here, because th e instrument does not make measurements over the Earth's surface. It is going to measure the solar irradiance through the Earth's atmosphere.

Table 25. Sunrise Sunset Scheduler Test: Configuration

#	Orbital Period secs	Altitude km	Data Type	Required Data Rate	Scheduler Onset Time secs
1	5760	566	0	1 Kbps	0
2	5760	566	0	50 Kbps	7480
3	5760	566	0	90 Mbps	2000

Table 26. Sunrise Sunset Scheduler Test: Results

#	ON Time secs	OFF Time secs
1	0	27.069
2	7480	7489.62
3	3835.06	3862.129

In the case of Test #1, the Onset time is set when the Instrument should go on immediately. For Test #2, the Onset time is set close to Daylight Stop time. It is used to see if the Scheduler sets the OFF time at Daylight Stop time. In the case of test #3, the Onset time is set befor e Daylight Start time. It is done to check whether the Scheduler schedules the on time on Daylight Start time and not before. The test results for both the UDP and TCP cases were as expected.

TCP Results:

Table 27. Sunrise Sunset Scheduler Test: TCP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	27.069	27	27.024	26.999	I Kbps	1 Kbps	1 Kbps
2	9.62	9	9.009	9.009	50 Kbps	50 Kbps	50 Kbps
3	27.069	27	27.083	27.085	90 Mbps	89.757 Mbps	89.750 Mbps

UDP Results:

Table 28. Sunrise Sunset Scheduler Test: UDP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	27.069	27	27.022	27.022	I Kbps	1 Kbps	1 Kbps
2	9.62	9	9	9	50 Kbps	50 Kbps	50 Kbps
3	27.069	27	27.103	27.108	90 Mbps	89.674 Mbps	89.659 Mbps

7.7 Tests done using other Traffic types

The following section contains results obtained when other traffic types were used.

7.7.1 Queued burst traffic (Qburst)

In the queued burst traffic mode, the hosts under test transmit data every some specific intervals, specified by the burst period used in the scripts. The advantage of this algorithm is that variations in available line rate will not cause it to miss blocks generated by interrupts arri ving before previous write completes. The drawback is that characteristics of the traffic are influenced by the queuing delay.

The following were the results obtained:

Test startup conditions:

Table 29. Queued Burst Traffic Test: Configuration

#	Scheduler ID	Orbital Period secs	Altitude km	Surface to turn ON	Data Type	Required Data Rate	Scheduler Onset Time Secs
1	0	5760	566	3	1	1 Mbps	0
2	1	5760	566	3	1	10 Mbps	2000
3	2	5760	566	3	1	50 Mbps	5000

Results:

Table 30. Queued Burst Traffic Test: Results

#	Surface turned ON	ON Time secs	OFF Time secs
1	3	1621.7	1645.71
2	3	3621.7	3645.71
3	3	7462.6	7489.62

<u>TCP Results:</u>

Table 31. Queued Burst Traffic Test: TCP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	23.9667	24	24.031	24.031	1 Mbps	999 Kbps	999 Kbps
2	NA	NA	NA	NA	10 Mbps	NA	NA
3	27.0688	27	27.051	27.056	50 Mbps	49.905 Mbps	49.896 Mbps

In test #2, the ON, OFF times were after daylight stop time (1727.11 secs) and before the d aylight start time (3862.13). The instrument was therefore not scheduled until the next daylight start time.

UDP Results:

Table 32. Queued Burst Traffic Test: UDP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	23.9667	24	24.045	24.046	1 Mbps	998 Kbps	998 Kbps
2	NA	NA	NA	NA	10 Mbps	NA	NA
3	27.0688	27	27.033	27.038	50 Mbps	49.939 Mbps	49.929 Mbps

7.7.2 Video traffic

Generally, there are two types of video traffic: (1) MPEG video stream, and (2) Teleconference video stream. Each of the video streams have been used for the test and the tests and results are explained below.

7.7.2.1 MPEG traffic

MPEG is VBR traffic. It is a video stream. Tests run using MPEG are as shown:

Test startup conditions:

Table 33. MPEG Traffic Test: Configuration

#	Scheduler ID	Orbital Period secs	Altitude km	Surface to turn ON	Data Type	Required Data Rate	Scheduler Onset Time secs
1	0	5760	566	3	2	640 Kbps	0
2	2	5760	566	3	2	400 Kbps	4000

The tests were run using the surface based scheduler (Scheduler ID 0) and the sunrise sunset scheduler (Scheduler ID 2).

Results:

Table 34. MPEG Traffic Test: Results

#	Surface turned ON	ON Time secs	OFF Time Secs
1	3	1621.7	1645.71
2	3	7462.6	7489.62

TCP Results:

Table 35. MPEG Traffic Test: TCP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	23.9667	24	31.401	31.401	640 Kbps	474 Kbps	474 Kbps
2	27.0688	27	29.431	29.433	400 Kbps	351 Kbps	351 Kbps

UDP Results:

Table 36. MPEG Traffic Test: UDP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	23.9667	24	31.404	31.405	640 Kbps	474 Kbps	474 Kbps
2	27.0688	27	29.435	29.436	400 Kbps	351 Kbps	351 Kbps

The results show that the MPEG throughput varies a lot from the required throughput. This is due to the fact that it is VBR traffic. Similar results were obtained when Video Teleconference traffic was used. Video Teleconference is also VBR traffic. The results are shown in the next section. In the case of the VBR traffic it can also be seen that the Actual duration of the data transfer is much higher than the Required duration (On Time), compared to the other traffic types.

7.7.2.2 Video Teleconference traffic

The tests have been carried out by scheduling the surface -based scheduler (Scheduler ID 0) and day night scheduler (Scheduler ID 1).

Test Conditions:

Table 37. Video Teleconference Traffic Test: Configuration

#	Scheduler ID	Orbital Period secs	Altitude km	Surface to turn ON	Data Type	Required Data Rate	Scheduler Onset Time secs
1	0	5760	566	3	3	500 Kbps	5000
2	1	5760	566	3	3	400 Kbps	4000

Results:

Table 38. Video Teleconference Traffic Test: Results

#	Surface turned ON	ON Time secs	OFF Time secs
1	3	351.51	455.368
2	3	5621.7	5645.71

<u>TCP Results:</u>

Table 39. Video Teleconference Traffic Test: TCP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	103.856	104	107.993	107.995	500 Kbps	489 Kbps	489 Kbps
2	23.9668	24	25.87	25.871	400 Kbps	371 Kbps	371 Kbps

<u>UDP Results:</u>

Table 40. Video Teleconference Traffic Test: UDP Results

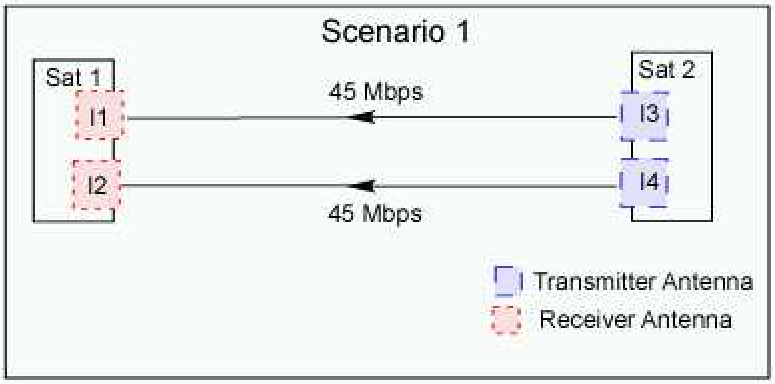
#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	103.856	104	107.998	107.999	500 Kbps	489 Kbps	489 Kbps
2	23.9668	24	25.867	25.868	400 Kbps	371 Kbps	371 Kbps

7.8 Test Scenario

The following scenarios simulate data transfer between multiple satellite instrument antennae.

7.8.1 Test Scenario 1

Figure 16. 2 Instrument Scenario



In the figure above, I1, I2, I3 and I4 are satellite antennae. The scenario is such that I3 sends burst traffic of 45 Mbps to I1 and I4 sends 45 Mbps qburst traffic to I2.

The results are tabulated below:

Test Conditions:

Table 41. Test Scenario 1: Configuration

#	Scheduler ID	Orbital Period secs	Altitude km	Surface to turn ON	Data Type	Required Data Rate	Scheduler Onset Time Secs
1	2	5760	566	NA	1	45 Mbps	0
2	2	5760	566	NA	1	45 Mbps	0

The scheduler in this case is the Sunrise S unset scheduler. Since it does not make measurements on the Earth's surface, the Surface to turn ON field is Not Applicable (NA).

Results:

Table 42. Test Scenario 1: Results

#	Surface turned ON	ON Time secs	OFF Time secs
1	NA	0	27.069
2	NA	0	27.069

<u>TCP Results:</u>

Table 43. Test Scenario 1: TCP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
2	27.069	27	27.055	27.056	45 Mbps	42.563 Mbps	42.561 Mbps
2	27.069	27	27.059	27.063	45 Mbps	39.513 Mbps	39.507 Mbps

<u>UDP Results:</u>

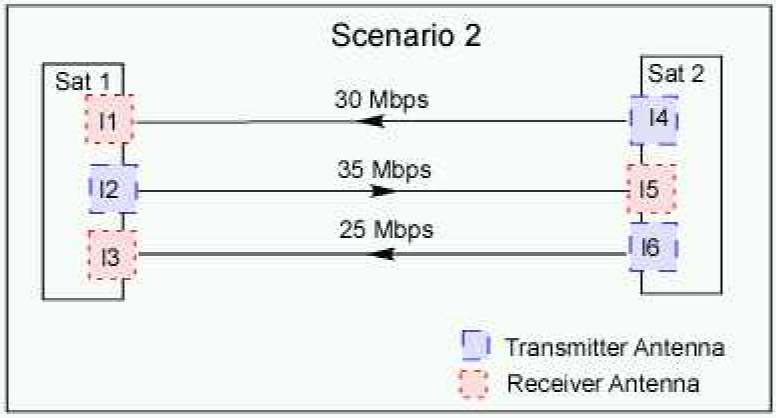
Table 44. Test Scenario 1: UDP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
2	27.069	27	27.017	27.019	45 Mbps	44.971 Mbps	44.951 Mbps
2	27.069	27	27.037	27.039	45 Mbps	44.938 Mbps	44.935 Mbps

It can be seen that the total traffic generated by UDP is greater than TCP traffic.

7.8.2 Test Scenario 2

Figure 17. 3 Instrument Scenario



Scenario 2 consists of data transfer among 6 satellite ins trument antennae. The results are tabulated below:

Test Conditions:

Table 45. Test Scenario 2: Configuration

#	Scheduler ID	Orbital Period secs	Altitude km	Surface to turn ON	Data Type	Required Data Rate	Scheduler Onset Time secs
1	0	5760	566	2	0	30 Mbps	0
2	2	5760	566	NA	1	35 Mbps	0
3	1	5760	566	3	1	25 Mbps	0

Results:

Table 46. Test Scenario 2: Results

#	Surface turned ON	ON Time secs	OFF Time secs
1	2	1350.1	1621.75
2	NA	0	27.069
3	3	1621.7	1645.71

TCP Results:

Table 47. Test Scenario 2: TCP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	271.623	271	302.349	302.354	30 Mbps	26.89 Mbps	26.89 Mbps
2	27.069	27	27.031	27.033	35 Mbps	34.96 Mbps	34.957 Mbps
3	23.9667	24	24.01	24.011	25 Mbps	24.99 Mbps	24.988 Mbps

UDP Results:

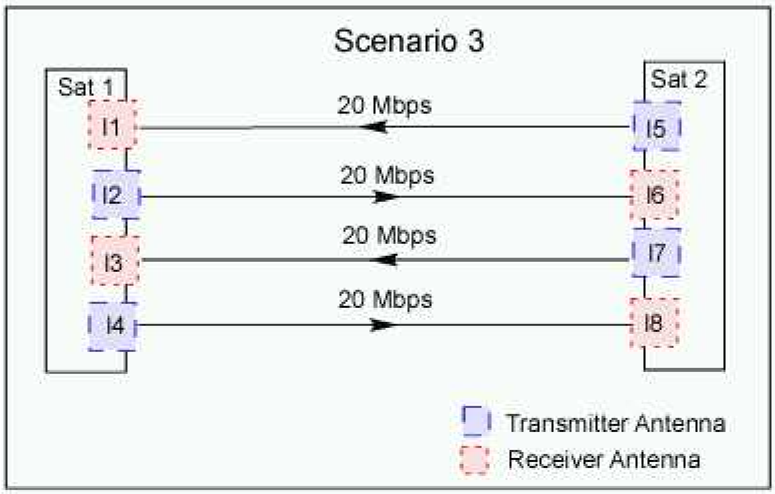
Table 48. Test Scenario 2: UDP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	271.623	271	271.211	271.211	30 Mbps	29.978 Mbps	29.943 Mbps
2	27.069	27	27.002	27.004	35 Mbps	34.998 Mbps	34.969 Mbps
3	23.9667	24	24.02	24.022	25 Mbps	24.979 Mbps	24.977 Mbps

7.8.3 Scenario 3

Scenario 3 involves 8 antennae, 4 of them transmitting and the other 4 receiving data.

Figure 18. 4 Instrument Scenario



Test Conditions

Table 49. Test Scenario 3: Configuration

#	Scheduler ID	Orbital Period Secs	Altitude km	Surface to turn ON	Data Type	Required Data Rate	Scheduler Onset Time secs
1	0	5760	566	2	0	20 Mbps	0
2	2	5760	566	NA	1	20 Mbps	0
3	1	5760	566	3	1	20 Mbps	0
4	1	5760	566	3	1	20 Mbps	0

Results

Table 50. Test Scenario 3: Results

#	Surface turned ON	ON Time secs	OFF Time secs
1	2	1350.1	1621.75
2	NA	0	27.069
3	3	1621.7	1645.71
4	3	1621.7	1645.71

<u>TCP Results:</u>

Table 51. Test Scenario 3: TCP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	271.623	271	303.739	303.977	20 Mbps	17.846 Mbps	17.832 Mbps
2	27.069	27	27.031	27.033	20 Mbps	19.977 Mbps	19.976 Mbps
3	23.9667	24	24.04	24.041	20 Mbps	19.967 Mbps	19.966 Mbps
4	23.9667	24	24.019	24.02	20 Mbps	19.984 Mbps	19.983 Mbps

<u>UDP Results:</u>

Table 52. Test Scenario 3: UDP Results

#	Duration in seconds				Throughput		
	Expected	Calculated	Actual TX	Actual RX	Expected	Actual TX	Actual RX
1	271.623	271	271.511	271.51	20 Mbps	19.963 Mbps	19.953 Mbps
2	27.069	27	27.001	27.002	20 Mbps	20 Mbps	19.769 Mbps
3	23.9667	24	24.029	24.031	20 Mbps	19.976 Mbps	19.975 Mbps
4	23.9667	24	24.02	24.022	20 Mbps	19.983 Mbps	19.632 Mbps

The 3 scenarios show that the scheduler schedules as expected and the required throughput is generated.

7.9 Evaluation

The schedulers have been tested using various conditions and the results have been analyzed. It is found that the results are same as expected result, except for the fact that the scheduled ON time is approximated to the nearest integer. Burst traffic and Qburst traffic have been found to give better throughput results and smaller variation in duration compared to VBR traffic types.

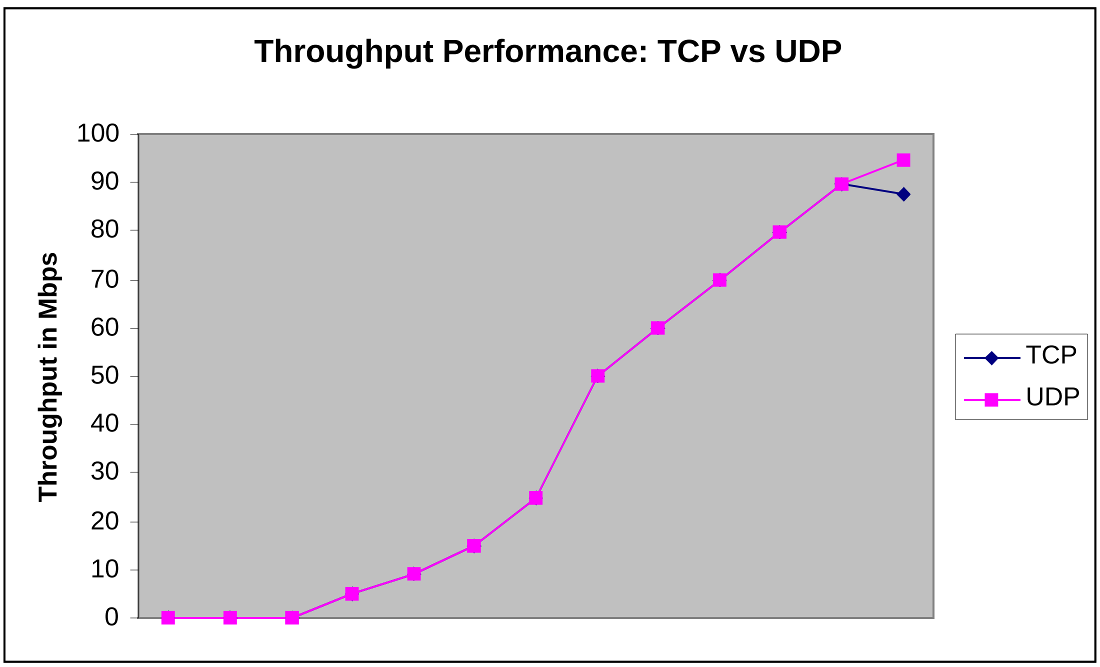
In the case of TCP, the maximum number of packets that can unacknowledged at any given time is constrained by the *Window size*. The *sliding window* concept makes stream transmission efficient. To, achieve reliability, the sender transmits a packet and then waits for an acknowledgement before transmitting another. The sliding window technique is a more complex form of positive acknowledgement and retransmission. Because a well-tuned sliding window protocol keeps the network completely saturated with packets, it obtains substantially higher throughput than a simple positive acknowledgement protocol. Increasing the window size increases the throughput, as a greater number of packets can be placed in the buffer unacknowledged.

The graph below shows the plotted throughput values for TCP and UDP:

Table 53. Throughput: TCP vs. UDP

Throughput in Mbps	
TCP	**UDP**
0.001	0.001
0.01	0.01
0.1	0.1
4.994	4.991
8.993	8.992
14.985	14.975
24.964	24.966
49.955	49.955
59.945	59.86
69.924	69.824
79.872	79.791
89.712	89.605
87.611	94.624

Figure 19. Throughput performance – TCP vs. UDP



UDP is found to give better throughput than TCP for throughput values greater than 70 Mbps. This is because, when the throughput is very high, TCP becomes slower because it has to wait for ack nowledgements and may need to retransmit if there is no acknowledgement received. UDP is connectionless. Therefore, it just keeps transmitting the packets, irrespective of whether the packet has reached the destination or has been lost.

7.10 Future Work

- Implementing the scheduler by obtaining satellite configurations directly from STK using STK Connect.
- Using a better topology resolution (The current Earth surface (Topology) database has a resolution of 0.5^0).

Appendix A

References

[1] Earth Science Enterprise, Detsination : Earth, http://www.earth.nasa.gov/

[2] NASA's Earth Observing System, http://eospso.gsfc.nasa.gov

[3] Landsat 7 Gateway http://landsat.gsfc.nasa.gov/

[4] Terra – The EOS Flagship http://terra.nasa.gov/

[5] TDRSS Overview, http://nmsp.gsfc.nasa.gov/tdrss/oview.html

[6] Minden, G.J., Evans J.B., *Architecture for Space Based Internets,* NASA ESE Proposal, December 2000

[7] Satellite Tool Kit, www.stk.com

[8] NetSpec

Roelof, Jonkman J. T., *NetSpec: Philosophy, Design and Implementation,* MS Thesis, University of Kansas, February 1998

Beng Ong Lee, *Wide Area ATM Network Experiments Using Emulated Traffic Sources*, University of Kansas, Lawrence, 1995

Beng Ong Lee, Victor S. Frost, Roelof Jonkman, *NetSpec 3.0 Source models for telnet, ftp, voice, video and WWW traffic*, Information and Telecommunication Technology Center, University of Kansas, ITTC-TR-10980-19 1997

NetSpec: A tool for Network Measurement & Experiment, www.ittc.ku.edu/netspec

[9] Earth Observation World, Fundamentals of Remote Sensing, http://www.eoc.nasda.go.jp/experience/rm_kiso/top_e.html

[10] Satellite Orbits

NASA Observatorium Education, Reference Module, Satellite Orbits, http://observe.ivv.nasa.gov/nasa/education/reference/orbits/orbits.html

"Looking at Earth From Space", a Teacher's Guide with Activities for Earth and Space Science, 1994, http://asd-www.larc.nasa.gov/SCOOL/orbits.html

[11] Landsat 7, Science Data Users Handbook, Chapter6 – Data Properties, Temporal Characteristics, http://ltpwww.gsfc.nasa.gov/IAS/handbook/handbook_htmls/chapter6/chapter6.html

[12] Michael D King, *24 EOS Measurements*, EOS Senior Project Scientist, March 14, 2000, http://eospso.gsfc.nasa.gov/ftp_docs/measurements.pdf

[13] Anatomy of a satellite, Communications – Overview, http://collections.ic.gc.ca/satellites/english/anatomy/comm/index.html

[14] SBI Node Software

Pooja J Wagh, *Space Based Internet Node Software Architecture Design*, Version 0.4, August 29, 2001, ITTC, University of Kansas

Leon S. Searl, *Space Based Internet Node Software Requirements*, Version 0.1,
August 29, 2001, ITTC, University of Kansas

[15] Virtual Ethernet (VETH)

Sandhya Rallapalli, *Space Based Internet Virtual Ethernet Interface Specifications*, Version 0.1, September 06, 2001, ITTC, University of Kansas

Sandhya Rallapalli, *Space Based Internet Virtual Ethernet Design*, Version 1.0, September 07, 2001, ITTC, University of Kansas.

[16] Emulation Manager

Sujit R. Baliga, *Space Based Internet Emulation Software Interface Specifications*, Version 0.1, March 23, 2001, ITTC, University of Kansas

Sujit R. Baliga & Sandhya Rallapalli, *Space Based Internet Emulation Software Architecture*, Version 1.1, September 4, 2001, ITTC, University of Kansas

[17] Operations Node Software
Sujit R. Baliga, *Space Based Internet Operations Node Software Design*, Version 0.1, November 20, 2001, ITTC, University of Kansas

[18] Operations Node Instrument Scheduler

Karthik N. Thyagarajan , *Space Based Internet Operations Node Instrument Scheduler Interface Specification*, Version 0.5, September 10, 2001, ITTC, University of Kansas

Karthik N. Thyagarajan , *Space Based Internet Operations Node Instrument Scheduler Design*, Version 0.5, August 29, 2001, ITTC, University of Kansas

[19] Node Instrument Scheduling Module

Karthik N. Thyagarajan , *Space Based Internet Satellite Node Instrument Scheduling Interface Specification*, Version 0.5, September 10, 2001, ITTC, University of Kansas

Karthik N. Thyagarajan , *Space Based Internet Satellite Node Instrument Scheduler Design*, Version 0.5, August 29, 2001, ITTC, University of Kansas

[20] Karthik N Thyagarajan, *Space Based Internet Earth Surface Query Interface Specification*, Version 0.1, June 4, 2001, ITTC, University of Kansas

[21] Exploration – Earth's Atmosphere
http://liftoff.msfc.nasa.gov/academy/space/atmosphere.html

EOS satellite instruments

Acrimsat

http://acrim.jpl.nasa.gov/

http://ccs.honeywell-tsi.com/psla/FY2000PSLAs/ACRIMSAT_PSLA_0303.pdf

1. ADEOS

http://www.eorc.nasda.go.jp/ADEOS/Project/#Mis

http://www.eorc.nasda.go.jp/ADEOS/Project/

http://www.nasda.go.jp/Home/Presskit/H-II_4_e/cht2-1_e.html

OCTS

http://www.eoc.nasda.go.jp/guide/satellite/sendata/octs_e.html

NSCAT

http://winds.jpl.nasa.gov/missions/nscat/nscatindex.html

2. ADEOS II

http://www.eorc.nasda.go.jp/ADEOS-II/

http://www-ilas2.nies.go.jp/eng/info/info_1.html

GLI

http://www.eoc.nasda.go.jp/guide/satellite/sendata/gli_e.html

http://titan.eorc.nasda.go.jp/test/GLI/gli.html

ILAS II

http://www-ilas2.nies.go.jp/eng/info/info_1.html

POLDER

http://www-projet.cst.cnes.fr:8060/POLDER/index.html

3. Cloudsat

http://cloudsat.atmos.colostate.edu/

4. EOS Chem (EOS Aura)

http://eos-chem.gsfc.nasa.gov/

HIRDLS

http://www.eos.ucar.edu/hirdls/

http://aura.gsfc.nasa.gov/hirdls/

MLS

http://aura.gsfc.nasa.gov/mls/

http://mls.jpl.nasa.gov/

OMI

http://aura.gsfc.nasa.gov/omi/

TES

http://aura.gsfc.nasa.gov/tes/

5. EO 1

http://eo1.gsfc.nasa.gov/miscPages/home.html

http://www.earth.nasa.gov/nra/archive/nra99oes01/nra99oes01.txt

ALI

http://eo1.gsfc.nasa.gov/Technology/ALIhome1.htm

http://iplab2out.sdstate.edu/eo1/Documents/ALI.PDF

AC

http://eo1.gsfc.nasa.gov/Technology/AtmosCorr.htm

http://eo1.gsfc.nasa.gov/Technology/Documents/AC-description.PDF

6. EOS PM (Aqua)

http://eos-pm.gsfc.nasa.gov/

http://www.dss.inpe.br/programas/hsb/ingl/index.html

AMSR/E

http://wwwghcc.msfc.nasa.gov/AMSR/html/amsr_desc.html

http://titan.eorc.nasda.go.jp/test/AMSR/amsr.html

http://titan.eorc.nasda.go.jp/test/AMSR/amsrspec.html#Performances

http://www.eorc.nasda.go.jp/ADEOS-II/AMSR/DOCS/amsrov.html

http://eospso.gsfc.nasa.gov/eos_homepage/Instruments/AMSR-E/

MODIS

http://ltpwww.gsfc.nasa.gov/MODIS/MODIS.html

http://fpga.gsfc.nasa.gov/asdp/fall97rep/fall97rep.htm

AMSU

http://orbit-net.nesdis.noaa.gov/crad/st/amsuclimate/amsu.html

AIRS

http://www-airs.jpl.nasa.gov/

HSB

http://www.dss.inpe.br/programas/hsb/ingl/index.html

http://www.dss.inpe.br/programas/hsb/ingl/Especifi/Default.htm

CERES

http://gcmd.gsfc.nasa.gov/announcements/ceres.html

7. ICESat

http://icesat.gsfc.nasa.gov/

GLAS

http://virl.gsfc.nasa.gov/glas/index.html

8. Jason 1

http://topex-www.jpl.nasa.gov/jason1/

http://ilrs.gsfc.nasa.gov/jason1.html

http://sirius-
ci.cst.cnes.fr:8090/HTML/information/publication/news/news7/escudier_uk.html

http://oceanworld.tamu.edu/students/jason1/jason2_sm_table.htm

http://aviso.jason.oceanobs.com/html/mission/orbite_uk.html

JMR

http://aviso.jason.oceanobs.com/html/mission/instruments/jmr_uk.html

9. LightSAR

http://www-radar.jpl.nasa.gov/ussar/

http://fst.jpl.nasa.gov/customers/lightsar.html

http://lightsar.jpl.nasa.gov/lightsar_archive/facts/97/

10. ML1-OTD

http://thunder.nsstc.nasa.gov/otd/

http://wwwghcc.msfc.nasa.gov/otd_old.html

http://people.cs.und.edu/~jgardner/otdorbit.html

11. Picasso-Cena

http://www-picasso-cena.larc.nasa.gov/picasso.html

LIDAR

http://www-lite.larc.nasa.gov/n_overview.html

12. POSEIDON 1 & 2

http://aviso.jason.oceanobs.com/html/mission/heritage_tp_uk.html

DORIS

http://aviso.jason.oceanobs.com/html/mission/instruments/doris_uk.html

LRA

http://aviso.jason.oceanobs.com/html/mission/instruments/lra_uk.html

13. QuikScat

http://winds.jpl.nasa.gov/missions/quikscat/quikindex.html

14. SAGE II

http://www-sage2.larc.nasa.gov/instrument/

http://eospso.gsfc.nasa.gov/eos_edu.pack/p37.html

15. SAGE III

http://www-sage3.larc.nasa.gov/solar/text/text-s3occultation.html

http://eospso.gsfc.nasa.gov/eos_homepage/Satellites/Sage/

16. Seastar

http://seawifs.gsfc.nasa.gov/SEAWIFS.html

http://msl.jpl.nasa.gov/QuickLooks/seastarQL.html

http://daac.gsfc.nasa.gov/CAMPAIGN_DOCS/BRS_SRVR/seawifs_daily_ops.html

SeaWIFS

http://www.iitk.ac.in/phy/phy101/seawifs.html

17. Seawinds 1A

http://www.eorc.nasda.go.jp/ADEOS-II/

http://winds.jpl.nasa.gov/missions/quikscat/quikindex.html#ground

18. SPARCLE

http://wwwghcc.msfc.nasa.gov/sparcle/

http://wwwssl.msfc.nasa.gov/newhome/headlines/essd14nov97_1.htm

19. SRTM

http://www.jpl.nasa.gov/srtm/

20. Terra

http://terra.nasa.gov/

http://shookweb.jpl.nasa.gov/validation/Calendar/terra.htm

ASTER

http://asterweb.jpl.nasa.gov/instrument/instrument.htm

http://www.science.aster.ersdac.or.jp/users/parte1/01-4.htm

CERES

http://gcmd.gsfc.nasa.gov/announcements/ceres.html

MISR

http://eos-am.gsfc.nasa.gov/misr.html

http://www-misr.jpl.nasa.gov/mission/ispatres.html

MODIS

http://ltpwww.gsfc.nasa.gov/MODIS/MODIS.html

http://fpga.gsfc.nasa.gov/asdp/fall97rep/fall97rep.htm

MOPITT

http://www.science.sp-agency.ca/J1-MOPITT(Eng).htm

http://www.atmosp.physics.utoronto.ca/MOPITT/overview.html

http://www.eos.ucar.edu/mopitt/dp_abstracts.txt

21. TRMM

http://daac.gsfc.nasa.gov/CAMPAIGN_DOCS/hydrology/AUTORESP/qx2d.html

http://trmm.gsfc.nasa.gov/

http://pao.gsfc.nasa.gov/gsfc/service/gallery/fact_sheets/earthsci/trmm.htm

http://eospso.gsfc.nasa.gov/eos_homepage/Instruments/LIS/

22. UARS

http://uarsfot08.gsfc.nasa.gov/

http://daac.gsfc.nasa.gov/CAMPAIGN_DOCS/UARS_project.html

SUSIM

http://wwwsolar.nrl.navy.mil/susim_uars.html

SOLSTICE

http://uarsfot08.gsfc.nasa.gov/UARS_INSTS/Obs_Inst_SOLSTICE.html

ACRIM II

http://coconut.oma.be/RadiometryPapers/COSPAR98.html

CLAES

http://svs.gsfc.nasa.gov/vis/a000000/a000800/a000810/

http://www.badc.rl.ac.uk/data/claes/

ISAMS

http://www.badc.rl.ac.uk/data/isams/isamshelp.html

MLS

http://mls.jpl.nasa.gov/index.shtml

HRDI

http://www.sprl.umich.edu/HRDI/hrdi_homepage.html

23. VCL

http://essp.gsfc.nasa.gov/vcl/

http://www.geog.umd.edu/vcl/

Other Web References

http://eospso.gsfc.nasa.gov/eos_homepage/description.html

http://eospso.gsfc.nasa.gov/ftp_docs/measurements.pdf

http://www.sti.nasa.gov/tto/spinoff1996/11.html

http://eospso.gsfc.nasa.gov

http://www.crseo.ucsb.edu/geos/111.html

http://www.eoc.nasda.go.jp/experience/rm_kiso/top_e.html

http://www.eoc.nasda.go.jp/experience/rm_kiso/whats_feature_e.html

http://www.eoc.nasda.go.jp/experience/rm_kiso/study_satellitedata_e.html

http://www.eoc.nasda.go.jp/experience/rm_kiso/mecha_howto_e.html

http://www.eoc.nasda.go.jp/experience/rm_kiso/satellit_type_orbit_e.html

http://ltpwww.gsfc.nasa.gov/IAS/handbook/handbook_htmls/chapter6/chapter6.html

http://samadhi.jpl.nasa.gov/msl/glossary.html

http://collections.ic.gc.ca/satellites/english/glossary/lr.html#plorb

http://collections.ic.gc.ca/satellites/english/anatomy/comm/index.html

http://nmsp.gsfc.nasa.gov/tdrss/oview.html

http://oea.larc.nasa.gov/PAIS/Satellites.html

http://observe.ivv.nasa.gov/nasa/education/reference/orbits/orbits.html

http://observe.ivv.nasa.gov/nasa/education/reference/orbits/orbit_sim.html

http://asd-www.larc.nasa.gov/SCOOL/orbits.html

http://nersp.nerdc.ufl.edu/~waltr/Satellites/#GEO

http://spaceboy.nasda.go.jp/Note/Eisei/e/EIs06b_e.html

http://alos.nasda.go.jp/2/orbit-e.html

http://daac.gsfc.nasa.gov/CAMPAIGN_DOCS/FTP_SITE/readmes/toms_daily.html

http://www.tbs-satellite.com/tse/online/

http://www.nationalacademies.org/ssb/smallsatappende.htm

http://daac.gsfc.nasa.gov/CAMPAIGN_DOCS/atmospheric_dynamics/ad_education/glossary.html http://www.ifigure.com/engineer/aero/aero.htm

http://astro.gmu.edu/classes/a10594/notes/l02/l02.html

http://nersp.nerdc.ufl.edu/~waltr/Satellites/#glossary

http://samadhi.jpl.nasa.gov/msl/glossary.html

http://eospso.gsfc.nasa.gov/eos_edu.pack/p37.html

http://solarsystem.nasa.gov/features/planets/earth/earth.html

http://www.solarviews.com/eng/terms.htm#astunit

http://userpages.umbc.edu/~cellis3/gloss2.html

Appendix B

Routines and Commands

- **OpNodeInstSched**

The following are the specifications for the command line to run the Instrument Scheduler Program. The Instrument Scheduling Module program has a command line interface with the following arguments:

OpNodeInstSched <char* SatFile> <char* NetspecFile> <char* OutputFile>
<int StartTime> <int SatID>

Where,

OpNodeInstSched is the Program Name.
SatFile is the Satellite file containing the Orbital parameters required to schedule the instrument.
NetspecFile is the file into which the Netspec code is generated (explained in the next chapter)
OutputFile is the file in which the results are generated
StartTime specifies the time in seconds at which the Scenario starts for that instrument
SatID is the satellite ID

- **ReadSatConfig**

OpNodeInstSched calls the ReadSatConfig() routine by passing the Satellite ID as an argument

ReadSatConfig <int SatID>

This routine reads the configuration file with “.config" ap pended to the SatID. The configuration file parameters were as explained above. These parameters are read by ReadSatConfig and stored in a structure.

The structure is defined as:

```
typedef struct {
        int SchedID      /* ID to be used for scheduling */
        int OrbP;        /* Orbital Period in seconds */
        int Altitude;    /* Satellite altitude above the Earth
                            surface in km */
        int ActOnSur;    /* Earth Surface Value (0,1,2) */
        int DataType;    /* Data Characteristic (0 to 5) */
        int Rate;        /* Throughput of the instrument in
      kilobits
                            per second */
        int PNum;        /* Port Number */
        char Source[11]; /* IP Address of Transmitting Antenna */
        char Sink[11];   /* IP Address of Receiving Antenna */
};
```

The data type / characteristic is an integer and the numbers are defined as the following traffic types:

0 – Burst

1 – Queued burst

2 - MPEG

3 – Video teleconferencing

4 – Voice

5 – World Wide Web

- **ReadSatFile**

OpNodeInstSched calls Re adSatFile() with the arguments being the satellite file and start time of the simulation for that satellite.

ReadSatFile <char* SatelliteFileName> <int Time>

- **ReadDaylightFile**

The scheduler on being started calls the routine ReadDaylightFile, which obtains the Daylight Start and Stop times.

ReadDaylightFile <char* FileName>

Where,

ReadDaylightFile is the name of the program

FileName is the name of the satellite file, obtained from STK containing the Daylight Start and Stop times for the satellite.

- **GetSurfaceType**

This subroutine takes the satellite co-ordinates and returns the surface value. It also checks for the legal or in-range values of the co-ordinates and does approximation if required.

GetSurfaceType <double latitude> <double longitude> <TopoType * surfaceType>

Where

latitude: is of type **double,** which specifies the latitude co-ordinate of the satellite position in degrees.
longitude: is of type **double,** which specifies the longitude co-ordinate of the satellite position in degrees.
TopoType: is an **enum** defined as

```
typedef enum { ICE, WATER, LAND } TopoType;
```

surfaceType: is of type TopoType defined above.

The various values that can be stored in surfaceType are 0 for Ice, 1 for Water and 2 for Land. This routine has to be called after *LoadSurfaceFile()* routine and shall return error otherwise.

Return Values

The subroutine has two return values of type **int**:

- **-1:** when an error occurs

  Possible Error Conditions

  When the latitude or longitude values are not "in range"

  When the `GetSurfaceType()` gets called before the `LoadSurfaceFile()` is called

  The type of error is stored in "**errno**"

- **0:** If it is successful in returning the surface value.

Legal or In-RangeValues

- Latitude: 90 degrees (NORTH) to –90 degrees (SOUTH)
- Longitude: 180 degrees (EAST) to –180 degrees (WEST)

Approximation

- The latitude and longitude values are approximated to the nearest 0.5 degree if they are not a multiple of 0.5 degrees
- If the value is w.25 or w.75 then it is approximated to w.0 and w.5 degrees respectively irrespective of the sign

  Where

  w is the integer value of the number.

- **LoadSurfaceFile**

This subroutine opens the surface database file passed as an argument and stores the surface values.

LoadSurfaceFile <char* FileName>

Where

FileName is of type **(char *)** which is the name of the surface database file whose values are to be stored in the array which is used for internal storage.

The array can be accessed by `GetSurfaceType()` routine to get the surface information.

This routine has to be called before `GetSurfaceType()` routine and shall return error otherwise.

Return Values:

This subroutine has two return values of type **int**:

-1: is returned on error

0: is returned on success

- **SurfaceScheduler**

The Surface Schedule r is called by OpNodeInstSched based on the SchedID. OpNodeInstSched calls routine SurfaceScheduler() routine with the following as arguments:

SurfaceScheduler <int z1> <TopoType surfaceType[ ]>
<float LatLonVal[ ][ ]> TopoType ActOnSurface>
<float SurfaceIgnoreTime> <float PrevOn> <float PrevOff>
<int thruput> <int portnum> <char *netspecfile>
<char *outputfile> <int mtype> <int SchedID>

Where,

z1 is internal to the program

surfaceType is an array of type TopoType where TopoType is an enum d efined as:

```
typedef enum { ICE, WATER, LAND } TopoType;
```

LatLonVal is a two dimensional array, which stores the time, latitude and longitude values as described in 3.

ActOnSurface is of type TopoType and stores the surface value where the instrument is turned ON.

OrbPeriod is the orbital period

SurfaceIgnoreTime is the total ON time up to which the instrument may not be turned on if set > 0. Default is 0.

PrevOn and PrevOff are internal to the program.

thruput is the total thruput of the satellite instrument

portnum is the port number to be used for transmitting or receiving data.

netspecfile is the to which the Netspec script is to be generated.

outputfile is the file to which the output / result is written onto.

mtype is the data type as described in 2.

SchedID is the ID used to identify the scheduler for the satellite instrument

- **DayNightScheduler**

The program is started on being called by OpNodeInstSched based on the SchedID. The Sunlight Start and Sunlight Stop times (in Epoch seconds) are passed to the Day / Night Scheduler along with Netspec parameters.

DayNightScheduler <int z1> <TopoType surfaceType[ ]> <float LatLonVal[ ][ ]>
<TopoType ActOnSurface> <float SurfaceIgnoreTime>
<float PrevOn> <float PrevOff> <int thruput> <int portnum>
<char *netspecfile> <char *outputfile> <int mtype>
<int SchedID>

Where,

z1 is internal to the program

surfaceType is an array of type TopoType where TopoType is an enum defined as:

typedef enum { ICE, WATER, LAND } TopoType;

LatLonVal is a two dimensional array, which stores the time, latitude and longitude values as described in 3.

ActOnSurface is of type TopoType and stores the surface value where the instrument is turned ON.

OrbPeriod is the orbital period

SurfaceIgnoreTime is the total ON time up to which the instrument may not be turned on if set > 0. Default is 0.

PrevOn and PrevOff are internal to the program.

thruput is the total thruput of the satellite instrument

portnum is the port number to be used for transmitting or receiving data.

netspecfile is the to which the Netspec script is to be generated.

outputfile is the file to which the output / result is written onto.

mtype is the data type as described in 2.

SchedID is the ID used to identify the scheduler for the satellite instrument

- **SunriseSunsetScheduler**

The program is called by the Operation Node Instrument Scheduler based on the SchedID of the instrument.

SunriseSunsetSchedule <float StartTime> <int thruput> <int portnum> <char *Netspecfile> <char *OutputFile> < int mtype>

Where,

SunriseSunsetSchedule is the name of the program

StartTime is the start time of the simulation in seconds.

thruput is the data rate for the instrument

portnum is the port number to be used by Netspec to send / receive data

Netspecfile is the name of the Netspec file to be written

Outputfile is the name of the file in which the result is made available

mtype is the data characteristic for that instrument

- **NodeInstSched** (also called NetspecWrapper)

NodeInstSched <int mtype> <int thruput> <int duration> <int portnum> <char *FileName> <char SourceIP[ ]> <char DestinationIP[ ]>

Each field is separated by a space. The following is a description of the valid values of each field.

mtype - This field contains an integer value that is used to determine the type of Netspec data model to use.
Thruput - Data rate of the instrument traffic in Kbps.
Duration - Length of time for the traffic to be generated in seconds.
Port - The IP port number for Netspec to use on the local host. The destination host will use Port+1.
FileName - The filename to use for the Netspec script that will be given to Netspec to generate the data traffic.
Source - The IP address on the satellite host to put in the Netspec script. This should be in Dotted IP form. ("192.168.72.2")
Sink - The IP address on the destination host to put in the Netspec script. This should be in Dotted IP form. ("192.168.72.2")

- **WriteNetspecFile**

This routine calls one of 6 d ifferent types of subroutines depending on the value of *DataType* received as shown in the table below:

Table 54. Data Type Values

DataType	TrafficType
0	Burst
1	Qburst
2	Mpeg
3	video teleconferencing
4	Voice
5	www

The appropriate routine is called to write out the Netspec file.

Depending on the DataType value the Program Name shall be one of the following:

- `burst_script_gen( )`
- `qburst_script_gen( )`
- `voice_script_gen( )`
- `mpeg_script_gen( )`
- `vidtel_script_gen( )`
- `www_script_gen( )`

Function call accepts the following arguments:

ProgramName <int Thruput> < int Duration> < int PortNumber>
< char *NetspecFileName> < char *SourceIPAddress>
< char *DestinationIPAddress>

Where,

Thruput is the desired data rate of the satellite instrument.

Duration is the duration of data gathering (traffic generation, in this case)

PortNumber is the port number through which the data traffic generated is to be sent.

NetspecFileName is the Netspec file that shall be used to generate the traffic.

SourceIPAddress is the IP address of the SBI Node that is generating the traffic and transmitting it.

DestinationIPAddress is the IP address of the SBI Node to which the data being generated is sent.

Error Messages

Error shall be generated if the routine is unable to open the Netspec file for writing

Return Value

The program returns

- 0 on success
- 1 on failure

- **RunNetspec**

The NodeInstSched calls this subroutine . It executes the Netspec file that is passed to it as an argument and generates data.

RunNetspec <char *NetspecFileName>

Where,

NetspecFilename is the name of the Netspec file that has been generated.

Error Messages

It generates error if

- The given input file name does not exist
- The Netspec script has parse error

Return Value

The program returns

- 0 on success
- 1 on failure

Satellite Files

- ### Configuration File

All Operation Node applications use the same configuration file to determine information need for communication among the applications. The `OpNodeIsntSched` application retrieves TCP IP socket por t numbers from the configuration file in order to connect to the Operations Module and the Node Attributes Module.

Example Configuration File

Instrument: Solar Ultraviolet Spectral Irradiance Monitor (SUSIM)

Satellite: Upper Atmosphere Research Satellite (UARS)

SchedulerID: 2

OrbitalPeriod: 5760

Altitude: 566

ActOnSurface: 3

DataType: 0

DataRate: 2

Port#: 45010

Source: 10.67.10.2

Destination: 10.67.10.1

The configuration file parameters are explained as follows:

- Scheduler ID is the ID used to id entify the scheduling mechanism for the satellite. In the case of the example above, scheduling is done based on Sunrise-Sunset
- Orbital Period – orbital period of the satellite in seconds
- Act On Surface – Surface type where the satellite instrument turns O N. In the example above, the instrument acts on both Land and Ocean. In this case, the ActOnSurface parameter is just any arbitrary integer value. ActOnSurface is necessary only when the satellite instrument has a scheduler ID which is either 1 or 2 (Measurement over only one surface type)
- Data Type – Data characteristic (example: bursty, full stream). For this instrument, the data characteristic is bursty traffic
- Data Rate – Data throughput rate of the instrument in Kbps.
- Port Number – Port number through which the data is to be sent or received
- Source – The IP Address of the transmitting antenna
- Destination – The IP Address of the receiving antenna

It should be noted that both the source and destination parameters may be discarded once the routing module is set up.

- **Satellite Position File**

Based on the SchedID, for an instrument requiring a Surface-Based scheduler or a Sunrise Sunset scheduler, ReadSatFile() reads the time, latitude, longitude values from the given start time up to one orbit starting from the given latitude and longitude values including the given start time and stores them in a global structure.

The structure is defined as follows:

```
typedef struct {
     float Time;        /* Time in seconds (epoch seconds) */
     float Lat;         /* Latitude in degrees */
     float Lon;         /* Longitude in degrees */
              };
```

Example Satellite position File

Satellite: Upper Atmosphere Research Satellite

A portion of the file is shown below:

The file shows the satellite orbital pos ition on 11/19/01 from 10:08 AM of the UARS satellite

Table 55. Satellite Position File

Time (EpSec)	Lat (deg)	Lon (deg)
0	-17.878	-177.414
7.98	-17.469	-177.148
15.96	-17.059	-176.884
23.94	-16.649	-176.62
31.92	-16.239	-176.358
39.9	-15.828	-176.097
47.88	-15.416	-175.838
55.86	-15.004	-175.579
63.84	-14.592	-175.322
71.82	-14.18	-175.066
79.8	-13.767	-174.81
87.78	-13.353	-174.556
95.76	-12.94	-174.303
103.74	-12.526	-174.051
111.72	-12.112	-173.799
119.7	-11.697	-173.549
127.68	-11.282	-173.299
135.66	-10.867	-173.05
143.64	-10.452	-172.802
151.62	-10.036	-172.555
159.6	-9.62	-172.308

- **Satellite Daylight File**

In the case of an instrument with SchedID for a satellite instrument measurement based on Day / Night scheduling, Read DaylightFile() reads the Sunlight Start and Stop Times, which are then stored into a structure.

```
typedef struct {
    int SunlightStartTime;    /* Sunlight Start Time in Epoch
                                 seconds */
    int SunlightStopTime;     /* Sunlight Start Time in Epoch
                                 seconds */
               };
```

Example Satellite daylight File

Satellite: Upper Atmosphere Research Satellite

The file shows the satellite orbital position on 11/19/01 from 10:08 AM of the UARS satellite.

Table 56. Satellite Daylight file

Sunlight Times (Secs)	
Start Time	**Stop Time**
0	1727.11
3862.13	7489.62
9624.5	13252.12
15386.85	19014.63
21149.18	24777.13
26911.5	30539.63
32673.8	36302.13
38436.09	42064.63
44198.36	47827.13
49960.61	53589.63
55722.85	59352.13
61485.07	65114.63
67247.27	70877.14
73009.46	76639.64
78771.63	82402.14
84533.79	86400

- **SurfaceTypeData**

SurfaceTypeData is the file in which the Earth surface values are stored. The data is stored in the following format:

Storage Format:

- The surface values are stored as follows :

  0: If Ice

  1: If Water

  2: If Vegetative Land

  3: If Arid Land

- The surface database file "SurfaceTypeData" is in ASCII format.
- Each line describes the surface type for different values on the same consistent latitude.

- Each adjacent character will be a 0.5^0 increment in longitude
  - Line 1 therefore will have co-ordinates starting from (90^0 N, -180^0 W) to (90^0 N, 180^0 E)
  - The surface values for the next successive latitude will be in the next line which means that it will start from (89.5^0 N, -180^0 W) to (89.5^0 N, -180^0 E)
  - The file begins with the surface value for the latitude longitude co-ordinates(90^0 N, -180^0 W) and steps up by 0.5^0 longitude up to 180^0 E and then steps down by 0.5^0 latitude and so on with the last surface value representing the coordinates (-90^0 S, 180^0 E)

Error Checks:

Valid values are "/" for Ice, "." for Water and "*" and "&" for Land. Any other value will generate error.

- **SurfaceType.h**

  SurfaceType.h is the include file which contains the signature of the above two functions and is to be included in the program tha t calls the programs. The file contains the following definitions:

- Definition for the **enum type "TopoType"** used to define the surface values as:

```
typedef enum { ICE, WATER, LAND } TopoType;
```

0: for Ice

1: for Water

2: for Land

- Definition for the error value assigned **int errorno**
- Definitions for the various routines routines:
- Definitions for the **errorno** values.

  19001: Error due to missing new line delimiter while reading the database file.

  19002: If the `GetSurfaceType()` routine has been called before the `SurfaceTypeData` file has been loaded

  19003: Invalid latitude co-ordinate

  19004: Invalid longitude co-ordinate

  19005: Too many arguments

  19006: Invalid character in the file

  19007: Invalid number of rows

It should be noted that the Latitude and longitude co -ordinates that are other than multiples of 0.5 do not have a surface value in the surface database. Therefore, values, which are not a multiple of 0.5, are approximated to the nearest 0.5 degree.

Example Test Results

Configuration File: 1234.config

SchedulerID: 0
OrbitalPeriod: 5760
Altitude: 566
ActOnSurface: 2
DataType: 0
DataRate: 5000
PortNumber: 45024
Source: 10.67.10.2
Destination: 10.67.10.1

The instrument was to be scheduled using the Earth Surface -based scheduler. The turn on surface is Vegetati ve Land. The data is of type burst and the throughput is 5 Mbps.

The Satellite position file of UARS satellite was used.

The program was started with the following arguments:

OpNodeInstSched <UARS-TimeLatLon> <5Mbps> <5Mbps-out> <0> <1234>
Where
UARS-TimeLatLon – satellite position file
5Mbps – Netspec file name
5Mbps-out – Netspec output file name
0 – Start Time
1234 – Satellite ID

The following was the output obtained:

It has been commented for better understanding.

```
/* BEGIN : OpNodeInstSched */
Calling ReadSatFile routine............

        /* Begin ReadSatConfig routine */
        The Satellite ID is : 1234
        The Satellite Configuration file is: 1234.config
        Able to Access file
        SchedID = 0
        OrbPeriod = 5760
        Altitude = 566
        ActOnSurface = 2
        MsgType = 0
        Thruput = 5000
        Port# = 45024
        Source = 10.67.10.2
        Destination = 10.67.10.1
        /* END ReadSatConfig routine */

Given Start Time = 0

        /* BEGIN ReadSatFile routine */
        Given Time  = 0.000000
        // Reads the various satellite positions from the satellite position file and stores
        the latitude and longitude values for up to 1 orbit in a structure
        /* END ReadSatFile routine */
```

Surface Type to act on = 2

/* BEGIN SurfaceScheduler */

/* BEGIN GetSurfaceType */

// Obtains the surface values for the various satellite positions for up to 1 orbit and stores it in a structure

/* END GetSurfaceType */

MessageType = 0

Restart Position Count = 203

ON = 1350.124878

OFF = 1621.747559 seconds

Current System Time Is: 1007560524

ON = 1007561874

OFF = 1007562145

Duration = 271

/*END SurfaceScheduler */

/* END OpNodeInstSched */

/* BEGIN NodeInstSched*/

//Calls the burst script generator routine

/*BEGIN burst_script_gen */

Thruput = 5000

blocksize = 6250.000000

//Writes out the Netspec file 5Mbps (shown at the end under files generated)

/* END burst_script_gen */

/* BEGIN RunNetspec */

Executing Netspec now........

/usr/local/bin/netspec /Karthik-Tests/InstSched/UDP/5Mbps

//The Netspec script is executed and the report is generated and written into the

file: 5Mbps-out

/* END RunNetspec*/

/*END NodeInstSched */

Files Generated:

- *5Mbps – Netspec script generated by burst_script_gen routine*

```
cluster{
	test 10.67.10.2 {
		type = burst (blocksize = 6250,period = 10000, duration = 271);
		protocol = udp;
		own = 10.67.10.2:45021;
		peer = 10.67.10.1:45022;
			}
	test 10.67.10.1 {
		type = sink (blocksize = 6250, duration = 271);
		protocol = udp;
		own = 10.67.10.1:45022;
		peer = 10.67.10.2:45021;
			}
}
```

- *Netspec Report obtained in file 5Mbps-out using RunNetspec routine*

NetSpec Version 3.0 alpha 1 Control Daemon bluenode02.sbidomain[1108] ready
NetSpec Version 3.0 TCPgamma 1, UDPalpha 1 Test Daemon
bluenode02.sbidomain[1109] ready
NetSpec Version 3.0 TCPgamma 1, UDPalpha 1 Test Daemon
bluenode01.sbidomain[1069] ready

NetSpec Control daemon, NetSpec Version 3.0 alpha 1

Report from bluenode02.sbidomain[1108] generated @ 12/05/01 07:10:49 CST

Test NetSpec Test daemon, NetSpec Version 3.0 TCPgamma 1, UDPalpha 1

Report from bluenode02.sbidomain[1109]

Start : 12/05/01 07:06:15 CST

End : 12/05/01 07:10:46 CST

Duration : 271.250 sec

Test data

Own address : 10.67.10.2:45021

Peer address : 10.67.10.1:45022

Type : burst

Blocks transmitted : 27101 blocks

Segments transmitted : 27101 segments

Bytes transmitted : 169381250 bytes

Thruput transmitted : 4.996 Mbps

Failed cycles : 0

Blocksize

Minimum : 6250 bytes

Maximum : 6250 bytes

Mean : 6250.000 bytes

Variance : 0.000 bytes

Deviation : 0.000 bytes

SegmentSize

Minimum : 6250 bytes

Maximum : 6250 bytes

Mean : 6250.000 bytes

Variance : 0.000 bytes

Deviation : 0.000 bytes

Repeat count

Minimum	: 1
Maximum	: 1
Mean	: 1.000
Variance	: 0.000
Deviation	: 0.000
Period	
Minimum	: 10000 usec
Maximum	: 10000 usec
Mean	: 10000.000 usec
Variance	: 0.000 usec
Deviation	: 0.000 usec

Protocol data (socket options)

Protocol	: UDP/IP
Socket type	: DGRAM
Transmit	
buffer	: 65535 bytes
low water	: 1 bytes
time out	: 0 usec
copy avoid	: False
Receive	
buffer	: 65535 bytes
low water	: 1 bytes
time out	: 0 usec
copy avoid	: False
Debug	: False
No routing	: False
Loopback	: False

Process stats (rusage)

System CPU time	: 0 us
User CPU time	: 0 us
Total CPU time	: 0 us
Page reclaims	: 1

Test NetSpec Test daemon, NetSpec Version 3.0 TCPgamma 1, UDPalpha 1

Report from bluenode01.sbidomain[1069]

Start	: 12/05/01 07:05:11 CST
End	: 12/05/01 07:09:42 CST
Duration	: 271.248 sec
Test data	
Own address	: 10.67.10.1:45022
Peer address	: 10.67.10.2:45021
Type	: sink
Segments received	: 27099 segments
Bytes received	: 169368750 bytes
Thruput received	: 4.995 Mbps
Segmentsize	
Minimum	: 6250 bytes
Maximum	: 6250 bytes
Mean	: 6250.000 bytes
Variance	: 0.000 bytes
Deviation	: 0.000 bytes

Protocol data (socket options)

Protocol	: UDP/IP
Socket type	: DGRAM
Transmit	
buffer	: 65535 bytes
low water	: 1 bytes
time out	: 0 usec
copy avoid	: False
Receive	
buffer	: 65535 bytes
low water	: 1 bytes
time out	: 0 usec
copy avoid	: False
Debug	: False
No routing	: False
Loopback	: False

Process stats (rusage)

System CPU time	: 0 us
User CPU time	: 10000 us
Total CPU time	: 10000 us
Page reclaims	: 1

Appendix C: Glossary

Abbreviations &Acronyms

Table 57. Abbreviations & Acronyms – Satellites

#	Acronym	Satellite
1	ADEOS	Advanced Earth Observing Satellite
2	ADEOS II	Advanced Earth Observing Satellite II
3	CloudSat	Cloud Satellite
4	EO 1	Earth Observing 1
5	EOS Chem	Earth Observing System Chem
6	EOS PM	Earth Observing System PM
7	Icesat	Ice, Cloud, and land Elevation Satellite
8	LightSAR	Lightweight Synthetic Aperture Radar
9	ML1-OTD	MicroLab I - Optical Transient Detector
10	QuikSCAT	Quick Scatterometer
11	SAGE II	Stratospheric Aerosol and Gas Experiment II
12	SAGE III	Stratospheric Aerosol and Gas Experiment III
13	SPARCLE	SPAce Readiness Coherent Lidar Experiment
14	SRTM	Shuttle Radar Topography Mission
15	TRMM	Tropical Rainfall Measuring Mission
16	UARS	Upper Atmosphere Research Satellite
17	VCL	Vegetation Canopy Lidar

Table 58. Abbreviations & Acronyms - Instruments

#	Acronym	Instrument
1	AVNIR	Advanced Visible and Near Infrared Radiometer
2	OCTS	Ocean Color & Temperature Scanner
3	ILAS	Improved Limb Atmospheric Spectrometer
4	RIS	Retroreflector in Space
5	TOMS	Total Ozone Mapping Spectrometer
6	NSCAT	NASA Scatterometer
7	POLDER	Polarization and Directionality of the Earth's Reflectance
8	IMG	Interferometric Monitor for Greenhouse Gases
9	AMSR	Advanced Microwave Scanning Radiometer
10	GLI	Global Land Imager
11	ILAS II	Improved Limb Atmospheric Spectrometer II
12	CPR	Cloud Profiling Radar
13	PABSI	Profiling Oxygen A-Band Spectrometer and Visible Imager
14	ALI	Advanced Land Imager
15	AC	Atmospheric Corrector
16	HI	Hyperion Instrument
17	HIRDLS	High Resolution Dynamics Limb Sounder
18	MLS	Microwave Limb Sounder
19	OMI	Ozone Monitoring Instrument
20	TES	Tropospheric Emission Spectrometer
21	AMSR/E	Advanced Microwave Scanning Radiometer-EOS
22	MODIS	Moderate Resolution Imaging Spectroradiometer
23	AMSU	Advanced Microwave Sounding Unit
24	AIRS	Atmospheric Infrared Sounder
25	HSB	Humidity Sounder for Brazil
26	CERES	Clouds and the Earth's Radiant Energy System
27	GLAS	Geoscience Laser Altimeter System
28	JMR	Jason Microwave Radiometer
29	OTD	Optical Transient Detector
30	Lidar	LIght Detection And Ranging

31	ABS	A-Band Spectrometer
32	IIR	Imaging infrared radiometer
33	WFC	Wide Field Camera
34	SAGE II	Stratospheric Aerosol and Gas Experiment II
35	SAGE III	Stratospheric Aerosol and Gas Experiment III
36	SeaWIFS	Sea-viewing Wide Field-of-view Sensor
37	VNIR	Visible and Near Infrared Radiometer
38	SWIR	Short Wave Infrared Radiometer
39	TIR	Thermal Infrared Radiometer
40	MISR	Multi-angle Imaging Spectro-Radiometer
41	MOPITT	Measurements of Pollution in the Troposphere
42	PR	Precipitation Radar
43	TMI	TRMM Microwave Imager
44	VIRS	Visible/infrared Radiometer
45	LIS	Lightning Imaging Sensor
46	SUSIM	Solar Ultraviolet Spectral Irradiance Monitor
47	SOLSTICE	SOLar STellar Irradiance Comparison Experiment
48	ACRIM II	Active Cavity Radiometer Irradiance Monitor II
49	CLAES	Cryogenic Limb Array Etalon Spectrometer
50	ISAMS	Improved Stratospheric and Mesospheric Sounder
51	MLS	Microwave Limb Sounder
52	HALOE	Halogen Occultation Experiment
53	HRDI	High Resolution Doppler Imager
54	WindII	WIND Imaging Interferometer
55	PEM	Particle Environment Monitor

Satellite terms & Definitions

Albedo: The ratio of reflected -to-incoming energy is called "Albedo" from the Latin word meaning whiteness. *The albedo is defined as the ratio of the solar flux scattered from the surface to the solar flux incident on the surface. It is a measure of the fractional amount of solar radiation absorbed by the surface.* Theoretically, albedo can range from unity (a surface that does not absorb any of the incident radiation at the wavelength(s) of interest) to zero (a "black" surface that absorbs all incident radiation at the wavelength(s) of interest).

Attitude: The position in space of a spacecraft or aircraft. A satellite's attitud e can be measured by the angle the satellite makes with the object it is orbiting, usually the Earth. Attitude determines the direction a satellite's instrument's face. The attitude of a satellite must be constantly maintained

Attitude control: Stabilizin g a satellite's attitude (direction) in its orbit. Attitude control can be done by spinning the satellite, or by having it remain stabilized in three axes using an internal gyroscope and thrusters

Ascending node (AM Orbit): That point at which a planet, p lanetoid, or comet crosses to the north side of the ecliptic; that point at which a satellite crosses to the north side of the equatorial plane of its primary. Also called northbound node. The opposite is **descending node** or southbound node.

Azimuth angle: Azimuth measured from 0° at the north or south reference direction clockwise or counterclockwise through 90° or 180°. Azimuth angle is labeled with the reference direction as a prefix and the direction of measurement from the reference direction as a suffix. Thus, azimuth angle S 144° W is 144° west of south, or azimuth 324°. When azimuth angle is measured through 180°, it is labeled N or S to agree with the latitude and E or W to agree with the meridian angle.

Backscatter: The scattering of radiant energy into the hemisphere of space bounded by a plane normal to the direction of the incident radiation and lying on the same side as the incident ray; the opposite of forward scatter. Atmospheric backward scatter depletes 6 to 9 percent of the incident solar beam before it reaches the earth's surface. In radar usage, backward scatter refers only to that radiation scattered at 180° to the direction of the incident wave.

Baud: A unit of signaling speed. The speed in bauds is the number of code elements per second. Baud is the number of signal level changes per second in a line, regardless of the information content of those signals.

DAAC: Distributed Active Archive Center, a repository for earth science data

Descending Orbit (PM Orbit): A polar orbit traversing from the north pole to the south pole. For the NOAA sun synchronous polar orbital satellites this orbit crosses the equator during nighttime hours.

Doppler Shift: The wavelength of light emitted by a moving object is shifted. This effect is called the Doppler shift. If the object is coming toward you, the light is shifted toward shorter wavelengths, blue shifted. If the object is going away from you, the light is shifted toward longer wavelengths, red shifted. The amount of shift is bigger if the emitting object is moving faster.

This is not normally noticed for light. But it is easy to observe for sound:

Blue shifted = higher frequency = higher pitch.

Red shifted = lower frequency = lower pitch.

In Astronomy:

The amount of shift is bigger if the emitting object is moving faster.

Thus if the original wavelength is known, speed can be deduced.

For a spectral line from an element that can be identified, the wavelength is known.

Thus for many astronomical objects we can tell

Is it coming toward us?

Is it going away?

If so, how fast is it moving toward us or away from us?

But note that this method does not tell us how fast it is moving sideways.

Duty Cycle: Duty cycle (or duty factor) is a measure of the fraction of the time a radar is transmitting. It is important because it relates to peak and average power in the determination of total energy output. This, in turn, ultimately effects the strength of the reflected signal as well as the required power supply capacity and cooling requirements of the transmitter.

Earth's Radiation Budget: The balance between incoming and outgoing energy is called the Earth's radiation budget.

Eccentricity: is a measure of how circular a satellite's orbit is. For a perfectly circular orbit the eccentricity is zero; elliptical orbits have eccentricities between zero and one. The higher the eccentricity, the more "squashed" the orbit is.

Gimbal:

1. A device with two mutually perpendicular and intersecting axes of rotation, thus giving free angular movement in two directions, on which an engine or other object may be mounted.
2. In a gyro, a support which provides the spin axis with a degree of freedom.
3. To move a reaction engine about on a gimbal so as to obtain pitching and yawing correction moments.

El Niño (Spanish for the Christ child): This is the most important climate phenomenon known on a multi-year time scale. The name El Niño has been used by Peruvian fishermen for a long time to identify a phenomenon of warming of the ocean surface waters, often occurring at the end of the year and lasting for several months and which induces significant climactic change over the entire world.

Green House Effect: If a cloud forms in a previously clear sky, the cold cloud top reduces the longwave emission to space, and energy is trapped beneath the cloud top. The trapped energy increases the temperature of the Earth's surface and atmosphere until the longwave emission to space once again balances the incoming absorbed shortwave radiation. This process is called "greenhouse effect" and, taken by itself, causes a heating of the Earth's climate.

Limb: The edge of the apparent disk of a celestial body, as of the sun.
Limb of the earth: The edge of the earth at the horizon.

Lagrangian Points: Much could be learned about space if only it were possible to suspend a satellite motionless in space, observing changes of magnetic fields and particle flows at one fixed spot. It cannot be done. To stay up and resist gravity, a satellite must be constantly on the move and must follow its prescribed orbit. The best we can do is station-keeping: for instance, the motion of a satellite in a synchronous orbit around the equator is matched by the rotation of the Earth below it, allowing it to permanently stay above the same equatorial spot.

Nadir: That point on the celestial sphere vertically below the observer, or 180 degrees from the zenith.

Occultation: This describes the moment at which one celestial body passes in front of another, temporarily obscuring it.

Pass: The period of time the satellite is within telemetry range of a data acquisition station.

Polarization: Orientation of the vibration pattern of light waves in a singular plane. A right-handed polarized wave is defined as one receding from the observer and radiated by an electric vector rotating clockwise in a fixed plane that is in front of the observer and at right angles to the direction of propagation of the wave in question. Left-handed polarization is the rotation in a counter-clockwise manner. This recommended definition of circular (or elliptical) polarization sense is according to that of the Institute of Radio Engineers. The definition of classical physics is exactly the opposite.

Scatterometer: A scatterometer is a microwave radar sensor used to measure the reflection or scattering effect produced while scanning the surface of the earth from an aircraft or a satellite.

Sea Ice: A general term for the seasonal ice that forms to cover large parts of the ocean in the winter, and that melts back in the summer. Sea ice includes grease ice, frail ice, pancake ice and pack ice.

Solar Flares: Bright eruptions form the sun's chromosphere. Compare prominence. Flares may appear within minutes and fade within an hour. They cover a wide range of intensity and size, and they tend to occur between sunspots or over their penumbrae. Flares are related to radio fadeouts and terrestrial magnetic disturbances. Flares eject high-energy protons, which present a serious hazard to men in unshielded spacecraft.

Spatial Resolution: The size of the smallest object recognizable using the detector.

Spectrograph: An instrument that spreads out the light gathered by a telescope so that it can be analyzed to determine many different properties of celestial objects. These properties include the chemical composition and abundance of different elements, temperature, radial velocity, rotational velocity, and magnetic fields.

Spectrometer: An instrument used to study the electromagnetic spectrum.

TT&C: An acronym for the satellite subsystem Telemetry, Tracking, and Control. TT&C refers to the brain of a satellite and its operating system. TT&C is the satellite's method for storing and analyzing the data it collects, and controlling its various systems. It also logs every activity of the satellite, receives information from the ground station, and takes care of any general upkeep, or "housekeeping", the satellite needs to do.