NASA/TM—2004-212299

Low-Earth-Orbit Satellite Internet Protocol Communications Concept and Design

Richard A. Slywczak
Glenn Research Center, Cleveland, Ohio

February 2004

The NASA STI Program Office . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the Lead Center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA's counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized databases, organizing and publishing research results . . . even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at ***http://www.sti.nasa.gov***

- E-mail your question via the Internet to help@sti.nasa.gov

- Fax your question to the NASA Access Help Desk at 301–621–0134

- Telephone the NASA Access Help Desk at 301–621–0390

- Write to:
  NASA Access Help Desk
  NASA Center for AeroSpace Information
  7121 Standard Drive
  Hanover, MD 21076

NASA/TM—2004-212299

Low-Earth-Orbit Satellite Internet Protocol Communications Concept and Design

Richard A. Slywczak
Glenn Research Center, Cleveland, Ohio

National Aeronautics and
Space Administration

Glenn Research Center

February 2004

Acknowledgments

Thanks go to Phil Paulson for generating the original idea of researching space-based architectures, to Will Ivancic and Mark Allman for their outstanding knowledge of TCP/IP, to Dave Carek for insights on security and VPNs, and to Cal Ramos for supporting this activity and reviewing the paper.

This report is a formal draft or working paper, intended to solicit comments and ideas from a technical peer group.

This report contains preliminary findings, subject to revision as analysis proceeds.

Available from

NASA Center for Aerospace Information
7121 Standard Drive
Hanover, MD 21076

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22100

Available electronically at http://gltrs.grc.nasa.gov

Low-Earth-Orbit Satellite Internet Protocol Communications Concept and Design

Richard A. Slywczak
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135

1.0 Summary

This report presents a design concept for a low-Earth-orbit end-to-end Internet-Protocol- (IP-) based mission. The goal is to maintain an up-to-date communications infrastructure that makes communications seamless with the protocols used in terrestrial computing. It is based on the premise that the use of IPs will permit greater interoperability while also reducing costs and providing users the ability to retrieve data directly from the satellite. However, implementing an IP-based solution also has a number of challenges, since wireless communications have different characteristics than wired communications. This report outlines the design of a low-Earth-orbit end-to-end IP-based mission; the ideas and concepts of "Space Internet" architectures and networks are beyond the scope of this document. The findings of this report show that an IP-based mission is plausible and would provide benefits to the user community, but the outstanding issues must be resolved before a design can be implemented.

2.0 Introduction

2.1 Overview

The focus of this report is to determine the feasibility of developing an end-to-end Internet-Protocol- (IP-) based satellite mission where all entities (end users, data systems, field sites, etc.) interact with the satellite by sending and receiving both commands and data. The overall goal is to make the satellite appear as a node on the Internet so that common Internet applications (e.g., File Transfer Protocol (FTP, ref. 1), Web browsers (refs. 2 and 3), Simple Network Management Protocol (SNMP, ref. 4), etc.) can interact with it. For many missions, NASA uses a protocol called Consultative Committee for Space Data Standards (CCSDS) Space Packet Protocol[1] to communicate with the satellites (http://www.ccsds.org). CCSDS is an international organization and its main purpose is for satellite communications; the Space Packet Protocol version of CCSDS does not integrate well with protocols used in terrestrial computing (ref. 5). Currently, data are collected in CCSDS packets, downloaded, and then wrapped in IP packets for transmission on the terrestrial Internet. Once they reach their destination, they must be unwrapped before processing and then repackaged back into IP packets for distribution (see fig. 1). Ideally, the data should be collected in IP packets and these packets should persist until the data system or end user receives them. The IP-based method has important benefits:

(1) It potentially reduces the number of facilities needed to retrieve and process the data by eliminating the intermediate stations that either transform or tunnel the data in IP packets.
(2) Knowledge about IP protocols is much more pervasive through the computing community, and finding qualified developers should be easy.
(3) The method should decrease the resources needed, the costs, and the development time because it uses an already existing set of protocols.

[1]There are several packet structures defined in the CCSDS Networks and Data Links specification. The older Version 1 CCSDS packet structure used in the Space Packet Protocol is not IP-based, but the current specification provides for IP-based structures. When this report references CCSDS, it is referencing the older Space Packet Protocol.

However, using IPs is not a simple solution because the wireless environment can have different characteristics than the wired environment (e.g., higher bit error rates, higher latency, and smaller bandwidth). The terrestrial version of Transmission Control Protocol/Internet Protocol (TCP/IP) was not meant to handle most of these cases and can either time out, not function efficiently, or not function at all. Changes to the terrestrial version of the TCP/IP stack may be required before a solution can be implemented. These factors become more severe as satellites get farther from Earth and, even though they may not be as severe in a low Earth orbit, they should still be considered.

This report shows the design of an end-to-end mission and where the various TCP/IP-based protocols[2] can be implemented. It divides the mission into two parts. First, we develop the satellite bus and show how IP protocols can be used for science instruments and health and welfare monitoring. Next, we classify the links needed to communicate between the ground and satellite and show the protocols needed for these links.

This report shows, in summary form, that a conceptual IP-based mission is feasible and that the requirements can be met by these protocols. However, there is still a need for research and development to increase the robustness and to solve the challenges associated with using IP protocols on wireless networks. Future work can expand on the information presented in this report. Current and future missions (e.g., constellations, space networks, sensor webs)—as well as those missions operating in the medium-Earth-orbit and geosynchronous-Earth-orbit ranges—will need solidly designed communication infrastructures. In addition, both space-qualified communication hardware and software will need to be developed and/or tested to support these missions.

2.2 Purpose

For the last 20 years, the Space Packet Protocol version of CCSDS has become the standard for NASA satellite communications, whereas the terrestrial environment has been standardizing on IP. The current practice is to encapsulate CCSDS data into IP-based packets for transmission over the Internet and then unpack them before processing. This report provides guidance for an end-to-end communications design that implements IP for data communications. This design is developed from a systems engineering perspective rather than a purely research perspective. Each component of the design is defined, and the candidate IP-based protocols are specified.

Using IP protocols end-to-end can offer potential benefits to NASA, but specific mission requirements will determine whether an Internet-like solution is realistic or cost effective. Terrestrial commercial-off-the-shelf standards applied to space applications do not necessarily equate to cost savings or increase reliability. Although there are many potential advantages to an Internet-like model, there are also many limitations because of the unique needs of space-based systems. These issues are identified and discussed in this report.

2.3 Scope

The focus of the report is the development of an end-to-end IP design for a generic low-Earth-orbit satellite mission that communicates directly with ground stations. The analysis section contains the basic design of an IP-based mission; it also contains background information for review. The report is divided into the following sections:

3.1 Reasons for an IP-Based Mission
3.2 Issues Concerning IP-Based Missions

[2]TCP/IP is a suite of protocols that includes both routing and transport protocols. Transmission Control Protocol (TCP) is only one of many reliable transport protocols.

In addition, there are three appendixes, as follows:

3.0 Analysis

This section provides rationale for an IP-based mission and describes the current communication infrastructure and challenges associated with using IP in space. In addition, it describes one potential spacecraft bus design concept and the associated IP-based communications infrastructure.

3.1 Reasons for an IP-Based Mission

NASA has been successfully using CCSDS as the communications mechanism for satellites, but the types of missions that NASA launches (or is planning to launch) have also been dramatically changing. The missions are becoming more complex and have to provide high-rate data delivery to users. Also, NASA is looking for the means to create more economical satellite missions. Changing the infrastructure provides the following benefits:

(1) *Integration among heterogeneous space platforms.* NASA is developing complex constellation missions, where a group of satellites will fly and take measurements in unity. In this arrangement, NASA might provide some of the satellites, but commercial industry or even foreign governments could provide others. To ease the integration among satellites, a widely used standard, such as IP, should be adopted.

(2) *Funding for new missions rather than maintaining infrastructure.* NASA has been using and maintaining CCSDS for satellite communications, but by adopting IP, NASA can let industry and academia play a major role in maintaining the communications infrastructure and can benefit from the results. As new improvements are discovered and developed, they will be added to the protocol stack and tested in the terrestrial world. Adding these changes into a flight-qualified TCP/IP stack should be easier than going through the complete detect-code-test cycle.

(3) *Simple access to platforms by users.* Using the TCP/IP Suite, general users (e.g., instrument scientists, science team members, etc.) will be able to retrieve data directly from the satellite. This will eliminate the need to use an intermediate system (e.g., a data processing system or archive). Users will be able to access the system using standard TCP/IP-based applications (e.g., FTP) to retrieve the data. This will be extremely beneficial to satisfying real-time processing requirements.

(4) *Addressing the challenges of networking in space.* Space provides a unique and challenging environment for communications and networking. The goal is to have networks of satellites that can communicate using IPs similar to those used in terrestrial networks. Before NASA can attain that goal, it needs to launch a single mission that is completely IP-based, and then problems can be addressed and solved before complex missions are developed.

(5) *Real-time data delivery.* With an IP design, scientists can use FTP (or other IP-based data transfer applications) to directly download data from a satellite in real time. Real-time processing would greatly benefit scientists by decreasing the time between data collection, and processing and analysis by the scientists.

(6) *Greater interoperability between platforms.* In the current infrastructure, data must pass through two intermediary data processing centers (see fig. 1) to be transformed from the CCSDS-based specification (created on the satellite) to an IP-based protocol (used in the terrestrial Internet). These centers add overhead by reformatting the packets to add or remove (and then re-add) packet headers. Using CCSDS requires specialty knowledge of how data are stored and formatted before processing. Using a common protocol throughout would help to simplify the data processing from the satellite through the ground networks.

(7) *Increase in security.* A major advantage of creating an IP-based architecture is the wealth of knowledge that exists regarding security. We have the ability to use hardware and software that promote security concepts (e.g., routers, firewalls, and virtual private networks, VPNs) to make the satellite as secure as possible. If protecting the data is the concern, then IP Security Protocol (IPSec, ref. 6) can be used to encrypt the data packets until they reach the end users.

3.2 Issues Concerning IP-Based Missions

Using standardized IP end-to-end may offer significant advantages and cost savings for the software development of both spacecraft and ground systems. In addition, these protocols will allow seamless interoperability with the ground systems. NASA can benefit from the extensive use and refinement that IP has seen in the terrestrial market. However, there are also problems associated with using IP, and these must be evaluated against the mission requirements to determine if they can be resolved.

In this section, challenges concerning an IP-based mission are discussed. Both positives and negatives are presented since most are multifaceted.

(1) *Standardization.* The network community regulates the changes to the TCP/IP Suite via the Internet Engineering Task Force (IETF). Developers can submit change proposals to the IETF, and after the community has a chance to comment on the changes, they can be approved and adopted. However, even if changes are approved, developers are not required to incorporate these into commercial stacks. In fact, there probably are no commercially developed stacks that are truly standards compliant. Thus, each TCP/IP stack implementation differs from every other and from the IETF standards (e.g., timer values, etc.).

(2) *Reduced costs.* Using IP for satellite missions should produce both budget and schedule savings. A wealth of knowledge already exists for developing TCP/IP-based applications, so there should be a knowledgeable workforce available. In addition, developers will not have to learn a new interface. Existing testing tools for TCP/IP-based programs will ease testing. In addition to development, maintenance costs should be reduced since all sectors (Government, industry, and academia) would work to improve the basic TCP/IP stacks. NASA should be able to more easily add these changes into a flight-qualified TCP/IP stack rather than going through the complete detect-code-test cycle.

(3) *Technology lag and obsolescence.* Since it takes a significant amount of time to develop and qualify radiation-hardened products, space hardware can lag 5 to 10 years behind the technology curve. NASA needs to consider, and potentially rework, the model to space-qualifying hardware and software so that changes can be integrated into missions sooner to take advantage of newer developments.

(4) *Flight qualification.* NASA does not have a space-flight-qualified TCP/IP stack; this must be considered before an end-to-end IP-based mission can fly. This would be a major activity, but once completed, the stack could be used on many missions. As the stack was upgraded, regression testing would have to be performed, which would be an additional cost. However, there should be some savings in two of the three parts of the detect-code-test cycle.

(5) *Quality of checksums.* When a data packet is received, three separate checksums must be validated before that packet can be accepted. The first is the IP header checksum; the second is the TCP checksum; and the third is a link layer checksum. Stone and Partridge (ref. 7) show that the checksum can fail, but the packet will be processed successfully. They indicate that between one packet in 10 billion and one packet in a few million will have an undetected error. Stone and Partridge recommend that, for critical applications, a custom cyclic redundancy check (CRC) should be applied at the application level.

(6) *IP Security.* Since there was a small initial user base for the Internet, network security was not given a high priority when the protocols were originally developed, but with the explosion of the Internet and the World Wide Web, security has become an important issue. Security has been retrofitted into the IP (e.g., IPSec), and with third party tools (e.g., firewalls and VPNs). However, security must be a priority for any satellite mission and incorporated from the beginning. In addition, surveillance of satellites using intrusion detection systems (IDS) must be ongoing to ensure that they are not compromised. Future work in security could include testing a VPN with encryption over a satellite link and determining which IDSs would work or require modification.

3.3 Current Satellite Communications Design

NASA has been safely and successfully launching satellites, but for satellite communications, NASA has been relying on a monolithic specification called CCSDS. CCSDS touts the same benefits as TCP/IP, which is to reduce recurring and nonrecurring costs along with reducing mission risk. However, with CCSDS, there are limited services that NASA can provide to the general user community to interactively work with the satellite. To illustrate how the existing communications infrastructure works, this section describes a typical mission. It is important to see how the data must get translated, after leaving the satellite and before reaching the end user, as represented in figure 1 (Paulsen, Phillip E.: Internet Protocols in Space: Is It Time To Change? Presentation given at the NASA Glenn Research Center, Cleveland, OH, on July 11, 2002).

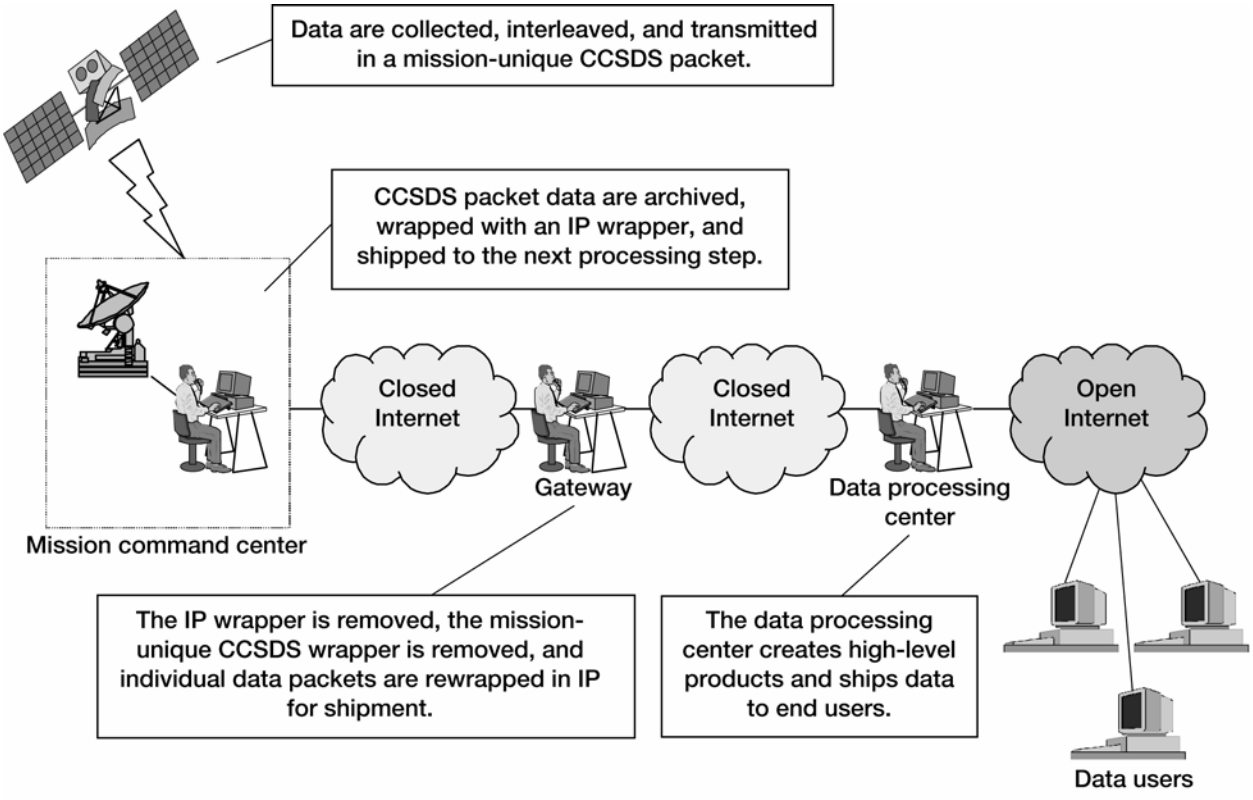

Figure 1.—Existing Consultative Committee for Space Data Standards (CCSDS) communications infrastructure.

Data transmitted from the satellite to end users via CCSDS take the following path (see fig. 1):

- *Step 1.* The data are collected on the satellite and placed in mission-specific CCSDS packets. These packets are defined and customized for that satellite mission. Step 1 takes place completely aboard the satellite.
- *Step 2.* The packets are transmitted to a ground terminal (for example, at White Sands, NM), which would receive the data directly from the satellite during a contact period.
- *Step 3.* Once the data have been initially downloaded, they are archived and then wrapped in IP packets that can be carried along a terrestrial network.
- *Step 4.* The data travel along a closed, private Internet until reaching a processing facility (for example, the Sensor Data Processing Facility at the NASA Goddard Space Flight Center) that is responsible for removing both the IP headers and the CCSDS headers. The resulting data, which are time corrected and checked for duplicates, are repackaged in TCP/IP packets for transmission.
- *Step 5.* The data arrive at the data processing system for the project, where they need to be calibrated, geolocated, and potentially, have higher-level products generated. At this point, individuals can access the data only if they have access to the closed Internet.
- *Step 6.* Scientists have to place orders with the processing system to get the data distributed to them. This distribution takes place over the open Internet via standard TCP/IP protocols.

Under the current communications infrastructure, it takes six steps to get the data from the satellite to the end users. By using an IP-based design, NASA should be able to reduce the number of steps, transmit the data to the users more easily, and in the process, reduce the number of processing centers that the data must pass through.

3.4 Conceptual IP-Compliant Design

The goal of this paper was to develop an end-to-end design that would extend from the spacecraft to the end users (e.g., the data processing system or scientists). The design can be divided into the following two components:

(1) *The satellite bus design.* This includes any networks or networking needed to collect and store data onboard the spacecraft. All data uploaded to the spacecraft will also be in IP packets and will pass along these same networks. An IP-compliant design will provide a standard interface between each of the components on the bus and the bus itself. Each component will be able to pass standard IP-compliant packets.

(2) *The links needed to transmit the data between the spacecraft and the end users on the ground.* We examine these links and also discuss and recommend the protocols to transfer data over these links.

3.4.1 Design considerations.—The end-to-end IP-compliant mission was designed from a system engineering perspective rather than a pure research perspective. The design started with a definition of the system and the flows that enter and exit it. This provided the top-level context diagram. The system was further divided into a second-level diagram by defining the systems that comprise the main system: the satellite mission.

The spacecraft entity is subdivided further in section 3.4.2. In this section, the design is composed of the typical instruments and subsystems that comprise a satellite bus and the associated networking components. The generic bus contains elements of a typical NASA satellite mission.

The remainder of the communications design describes the types of links for data transmission (section 3.4.4). As projects design their missions, they can tailor the links to meet their mission requirements.

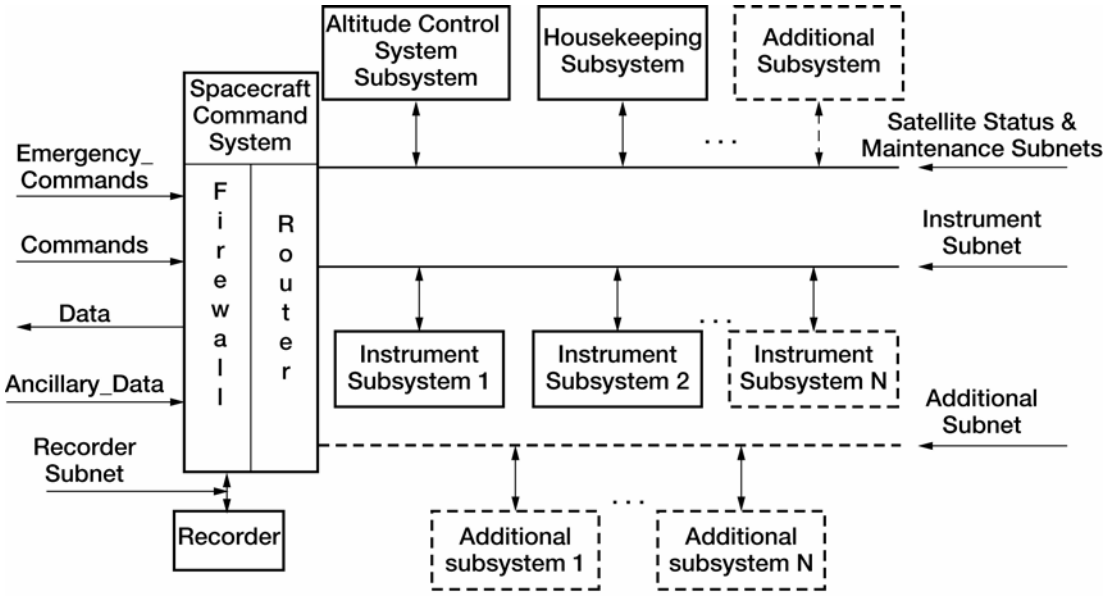

Figure 2.—IP-compliant satellite bus design.

3.4.2 IP-Compliant Satellite Bus Design.—The satellite bus (fig. 2) includes command and control modules, satellite housekeeping modules, science instruments, and data recorders; it also provides a standard interface to connect each of these components. The bus can be customized to meet the specific requirements of the mission.

Two main goals were considered in designing the spacecraft bus: we needed to provide IP compatibility throughout the bus, and we needed to make security a major design goal. This design takes both of these factors into consideration.

The bus is composed of three or more separate subnets, the Satellite Status & Maintenance Subnet, the Instrument Subnet, the Recorder Subnet, and any additional subnets needed by the mission. The firewall provides protection for these networks and the router transfers data to the appropriate subnet. For authentication, access control, and potentially encryption, the mission can implement a VPN.

4.4.2.1 Satellite bus network: The satellite bus will comprise at least three subnetworks that will be protected behind the router/firewall. Before anyone can gain access to any of the subnets, they will have to be validated at the firewall. The network on the satellite will be divided into at least three individual networks (or subnets):

- *Satellite Status & Maintenance Subnet.* This subnet will carry data for maintaining the health of the satellite, and satellite commands from the ground. For example, figure 2 shows two main modules connected to this subnet: the Attitude Control System Subsystem and the Housekeeping Subsystem and one or more additional subsystems. The additional subsystems show that other modules can be connected to this network, if required by the mission.
- *Instrument Subnet.* This subnet is where the science instruments reside. The number of instruments that can be connected to the bus is restricted by either the requirements of the mission or the physical limitations of the bus.
- *Recorder Subnet.* This subnet contains the data recorder that will store data for the command and science instruments. The data recorder is placed on its own subnet to simplify the connections, since each of the other subnets will need to send data to it.
- *Additional Subnet.* This represents one or more additional subnets that may be needed for specific missions. Additional subnets could be added for the logical division of instruments (command and/or science) or for security purposes. Limitations on the number of subnets should be based on either the number of networks that can be supported by the router or the power requirements of the satellite.

The satellite bus could have been designed a number of ways; the rationale for using multiple subnets follows:

- *Reduction of data traffic.* The subnets will separate the traffic for the command and control functions and the instrument collection duties of the satellite. This will allow for fewer collisions between these data because they will be on different physical networks.
- *Promotion of security on the satellite.* Using multiple subnets on the spacecraft will help to promote security by keeping the command/control and the instrument traffic on separate networks. For example, if an instrument scientist uploaded commands to the spacecraft, the commands would not traverse the Satellite Status & Maintenance Subnet; therefore, the scientist would not be able to send damaging commands to the satellite.

All subnets have access to a data recorder that will store all data from the science and control modules. The data recorder is a passive device that is connected to the router via another subnet (i.e., the Recorder Subnet). After the instruments collect the data, the data will be sent to the data recorder via the router. The data recorder will become an IP-addressable device that stores the data until information is requested to be transmitted to a ground terminal.

In this design all instruments and modules are IP compatible and able to plug directly into the bus using space-qualified connectors to an Ethernet interface. The protocol for the satellite bus will be standard TCP/IP. The reasons for this follow:

- *Latency is not a factor.* One problem with data transmission over wireless networks is latency. The latency is compounded by the overhead involved with TCP/IP (e.g., slow start, congestion control, etc.). But since all communications are restricted to the local network aboard the satellite, TCP/IP is an acceptable protocol.
- *Reliable data communications are required.* The data measured by the instrument are collected in real time and written to the recorder. The modules have no, or limited, methods of recreating or caching the data. Therefore, there has to be a reliable way of getting the data from the instruments to the recorder; TCP/IP will provide the reliable data transfer.

3.4.2.2 Router/firewall: The second part of the bus is composed of the firewall and router, which function as the interface between the local onboard network and the ground. All communications with the satellite will be required to pass through this interface before being passed to any module on the satellite.

The firewall will provide a layer of security by filtering the traffic being sent to the satellite. It will validate the network packets on the basis of a series of conditions or rules and will be able to either accept a packet for processing or deny the packet and drop it. If the packet passes the firewall rules, it will be sent to the router. Essentially, the router will provide the same basic functions as a terrestrial router. The main function will be to place the received packet onto the correct subnet so that the appropriate module can process it. A router will only be needed in the design when there are multiple subnets. If the mission decides on one network (i.e., a single network containing both the science and command and control instruments), then the packet would be validated by the firewall and placed on the network.

3.4.3 Emergency commanding.—During the mission, the satellite can become unstable for a number of reasons:

(1) The satellite could start tumbling in its orbit, making the main antenna ineffective for receiving commands from the ground.

(2) The network interface card (NIC) could become inoperable and not able to process the network packets.

During these instances, the ground will not be able to talk to the satellite and it will become uncommandable. Without some type of backup, the entire mission could be in jeopardy of being lost.

As a solution to this problem, the design (see fig. 2) allows for emergency commanding by connecting the bus to a low-rate modem and an omnidirectional antenna. This low-bit-rate connection would be used for simple commands to query and/or stabilize the satellite. Once the satellite was stabilized, commanding could resume using the normal interface.

To receive the signals, the spacecraft will need to include a multidirectional antenna, such as an omni, that can receive commands regardless of the orientation of the satellite. The commands will still pass through the firewall so that they can be validated before being processed by the commanding system. With emergency commanding, only the vital components of the satellite, such as the command and control and housekeeping systems, will be able to be manipulated. Other components, such as the science instruments, will likely be in standby mode through anomaly-correction schemes.

3.4.4 Satellite Link Descriptions.—The previous section presented a satellite bus design that allows for any information transmitted on the bus to be in standard IP-based packets. The next section describes the rest of the design in the end-to-end mission and the communication protocols that exist between them. The generic mission is divided into a set of links for transmitting data between the satellite and some entity (see fig. 3). The data flows to each entity are described, and a recommendation for one or more appropriate protocols is made for each link.

In figure 3, the entities are defined as follows:

- *Satellite*. The satellite will be responsible for hosting the science instruments as well as providing command and control functions. It will also be responsible for receiving commands from the ground and executing the commands, once they are validated. For more information, see section 3.4.2.
- *Data system.* The data system will be the main user of the science and command data collected by the spacecraft; it also will be responsible for uplinking command data to the satellite. It could consist of multiple organizations, such as a data collection facility, a data processing and distribution facility, and a mission operations center, or it could be a single organization. The design and function of a data system will be left to the specific mission. For simplicity in this document, they are combined into one entity.

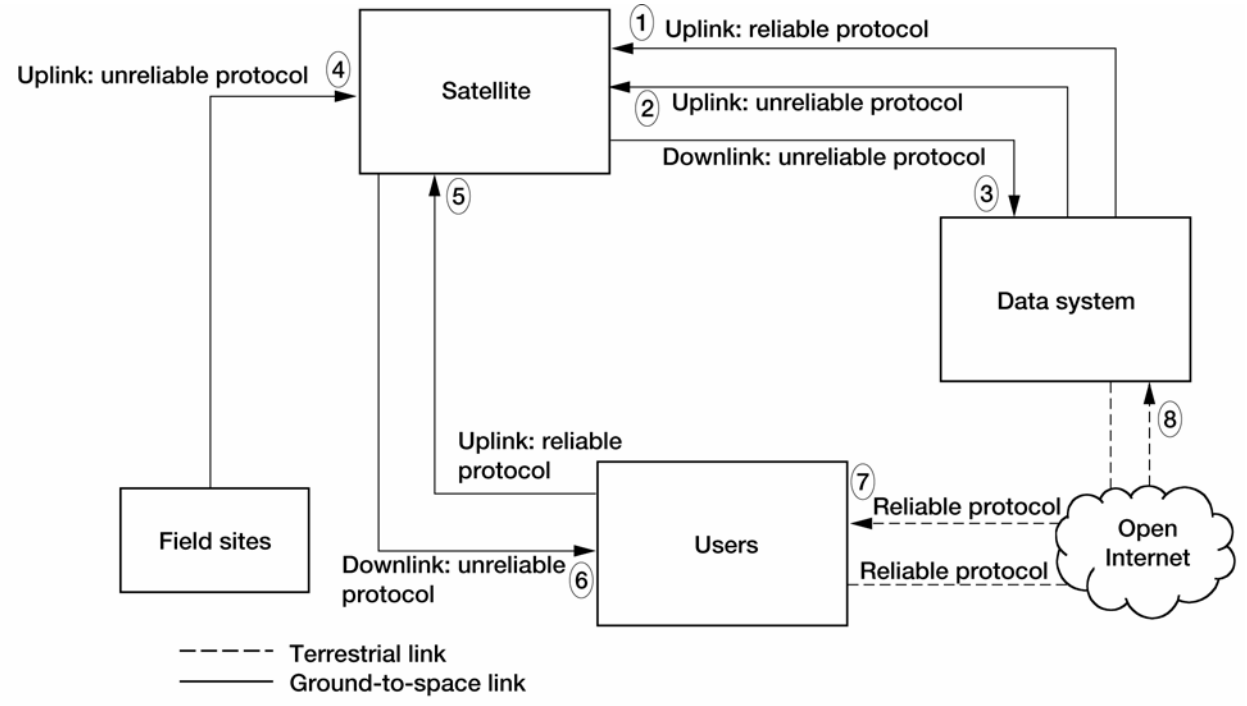

Figure 3.—Satellite data transfer link design.

- *Users.* This represents a set of data users that will retrieve data directly from the satellite. The mission will have the responsibility of defining those individuals that are part of this group, but the group will, at a minimum, consist of instrument scientists and researchers. The data would not be processed beyond what was performed as part of the onboard processing. These users would access the data via standard or standardlike TCP/IP applications, such as FTP.
- *Field sites.* These represent entities that will potentially need to uplink ancillary data to the satellite. These entities could be part of the data system or users, but are not required to be part of either of those groups. The ancillary data could represent calibration coefficients that will have to be applied to the data, or instrument parameters that are provided by the instrument scientists. A mission may not have a requirement for field sites.

Figure 3 shows the direction of the data flow and the type of protocol that must traverse each link. In this section, the links between each of the entities—identified by item number—are examined by their purpose and suggested protocols.

(1) *Commands—uplink, reliable protocol (data system to satellite).*

Description: These commands refer to the satellite's operating instructions that will be uplinked to the satellite. They will command both the satellite, via the Satellite Status & Maintenance Subnet, and the instruments, via the Instrument Subnet. The commands, which will be sent by the mission operations staff, will have to be validated and encrypted, since they could include mission-ending commands.

Protocols: The commands will be extremely important to the success of the mission and will have to be received and executed successfully. These files will have the additional benefit that they are small in size and will not take long to uplink. The best option is TCP/IP, since reliability and correctness are paramount. Since these commands can render the satellite unusable, the mission may want to consider an application-level CRC as an additional validity check.

(2) *Ancillary_Data— uplink, unreliable protocol (data system to satellite).*

Description: This is a data file that may be needed for onboard processing. Ancillary_Data could be either data needed by the instruments, such as calibration data, or data needed by any calibration/ transformation algorithms. The Ancillary_Data file will be uplinked from the Mission Operations staff.

Protocols: There are two options for uplinking Ancillary_Data: (1) The Ancillary_Data will have to be uplinked to the satellite, but this information will not be critical. The requirement should be to satisfy a predetermined processing schedule. Therefore, an inherently unreliable protocol with a built-in acknowledgment scheme could be used to transfer these data to the satellite. Some possible candidates would be Space Communications Protocol Standard's (SCPS's) TCP with Selective Negative Acknowledgments (SNACKS), and Multicast Dissemination Protocol (MDP). (2) If the size of the Ancillary_Data file was small and the contact time with the satellite was sufficient, a reliable protocol could be used (e.g., TCP/IP or Stream Control Transmission Protocol (SCTP)).

(3) *Data—downlink, unreliable protocol (satellite to data system).*

Description: These are a combination of raw science data collected by the instruments, data processed by onboard processors, and/or satellite health and welfare data. All of these data will be downloaded to the data system, which as specified earlier, can be one or more organizations. These are large files that can take some time to download, so they might need to be retrieved in chunks on successive passes.

Protocols: Similar to the Ancillary_Data mentioned in item 2, these data will have to be downloaded to the data system, but they will not be time critical, other than to meet processing deadlines. However, the file sizes could be very large, so a protocol that has a high degree of reliability (e.g., TCP/IP) could be inefficient given the short contact time. A more efficient method would be to use a rate-based protocol to send as much of the data as possible and then determine if all the chunks were sent successfully. Possible protocols for this download would be TCP with SNACKS, MDP, or SCTP.

(4) *Ancillary_Data—uplink, unreliable protocol (field sites to satellite).*

Description: These data are similar to the Ancillary_Data defined in item 2, except that the flow will be to the satellite from the Field_Sites. The field sites would need to uplink any data to the satellite that are needed for onboard processing.

Protocols: Similar to the Ancillary_Data uplinked by the data processing system, the designers could have their choice of either an unreliable protocol or, if the filesizes are small, a reliable protocol. The same options as specified in item 2 exist for this link.

(5) *Subset_Commands—uplink, reliable protocol (users to satellite).*

Description: Subset_Commands will be sent to the satellite by instrument scientists to control their instruments. The scientists will be responsible for uplinking the commands directly to the satellite, and the satellite will validate the data packets before sending them for processing. The commands will only traverse the Instrument Subnet after being validated by the firewall. If required by the mission, an application-level CRC could be used as an extra validation check for critical commands.

Protocols: These Subset_Commands are important because they have the potential of disabling the instrument. They will have to be uploaded successfully and, therefore, will need a reliable protocol. The Subset_Commands are small in size and, therefore, would not take too long to uplink. The best option is TCP/IP.

(6) *Data—downlink, unreliable protocol (satellite to users).*

Description: This is similar to the data flow defined in item 3, except this flow is between the spacecraft and the users. This data flow allows users to download data directly from the satellite to their home platforms, allowing them to retrieve the data in near real time rather than having to go through the data processing system. Depending on the capabilities of the spacecraft, scientists may have the option of downloading data that have been processed aboard the satellite.

Protocols: Unlike the previous data flows, this one should be a strictly rate-based protocol because science users will not want to sacrifice speed so that data are downloaded correctly the first time. The preference will be to have a faster download and then have any errors cleaned up after the download has finished. Giving the user community direct access to the satellite would benefit real-time processing schedules. The best protocol to use would be either TCP with SNACKS, MDP, or SCTP.

(7) *Processed_Science_Data—reliable protocols (data system to users).*

Description: Processed_Science_Data are the results of the orders placed by authorized users. Processed_Science_Data could be transferred via the network, but if the size exceeded an upper bound for network transfer, these data could also be transferred via tape.

Protocols: Since this will be a terrestrial link, the most efficient way to transfer the data will be via a reliable protocol, TCP/IP.

(8) *Requests—reliable protocols (users to data system).*

Description: Requests are orders that will be submitted by scientists for data that were processed at the Data_System. This is a terrestrial link, since both the users and the Data_System will be on a terrestrial link.

Protocols: Since this will be a terrestrial link, communication should be through a reliable protocol such as TCP/IP.

3.5 Satellite Security

One of the important challenges concerning computers and networking is security. There have been a significant number of attempts (and successes) at breaking into networks, Web sites, and computers in both the Government and commercial sectors. One of the challenges with creating an IP-based mission is that, although the protocol is well understood, its deficiencies are also well known and that makes securing networks difficult. Without proactive measures, the mission is vulnerable to an attack that could cripple it or cause a major catastrophe. It is important to protect the most important asset, and in this case,

it is the satellite since it is only accessible via commands after launch. The design implements a few measures that will make attacking the satellite much more difficult:

- *Virtual Private Network (ref. 8).* A VPN should be implemented to transmit critical or sensitive data safely from a source (e.g., an instrument scientist or a mission operations center) to the satellite. It would provide safe, secure, and private networking built on top of publicly accessible networks (e.g., the Internet). Using a VPN will permit commands and data to be sent securely to the satellite. VPNs will provide the following benefits:

    - *Authentication:* The VPN will ensure that the data originated at the source that is specified in the header.
    - *Access control:* The VPN will prevent unauthorized users from accessing the network.
    - *Confidentiality:* The VPN will prevent unauthorized users from reading data in transit across the network.
    - *Data integrity:* The VPN will prevent anyone from tampering with the data in transit across the network.

- *Firewall.* There needs to be some type of mechanism on the satellite to process commands and determine if they are valid and originated from a reputable source. The firewall will only allow traffic into the satellite that meets the filtering requirements, and it will drop those packets that do not comply with the filter. Some of these schemes will be based on port number, IP address, and other parameters. Although it is not an absolute measure, a firewall provides some defense against attackers.
- *Router.* The router will ensure that the appropriate data get placed on the correct subnet according to the IP addresses in the packet. The router should prevent rogue packets from getting to the instruments, since they would have to have IP addresses similar to those of the real instruments. Although this is not too difficult, a router adds an additional line of security when coupled with a firewall.
- *Separate networks.* The spacecraft and the science instrument will be commanded via two different networks. When commands are sent to the satellite, usually by mission operations, they will traverse only one segment of the network (i.e., the Satellite Status & Maintenance Subnet) and the commands sent by the instrument scientists will only traverse the Instrument Subnet. This should ensure that no one outside of mission operations will be able to send mission-ending commands to the spacecraft, once it has passed through the VPN, firewall, and router.

This design takes security into account and offers a proactive approach to ensure that the spacecraft will survive an attack. This section covers only the spacecraft, but these same principles could be applied to the ground networks (e.g., the data processing system). Most importantly, ground networks are much easier to fix than assets in space.

3.6 Encryption of Data Packets

As described in the last section, a VPN provides for data encryption to prevent unauthorized users from reading or modifying the data. Although it is expected that attackers will mean to harm the satellite (or some resource), they may be interested in only eavesdropping on either the communications or data and commands that traverse between the ground terminals and the satellite. This is a much tougher problem to handle, since the data may travel on the open Internet, and most likely, through networks not owned or managed by NASA. Because of open or public networks, NASA cannot mandate who has access, what access privileges those people have, or who has the right to connect to those networks. Encryption, as part of a VPN, should be used for critical commands to ensure their safety.

Eavesdropping is a problem, since the packets by default are sent “in the clear” and are not encrypted. Existing technologies—such as divert sockets, raw sockets (on some platforms), and packet filtering—can allow end users to see the packet data that are traversing the network. Applications that use these technologies are tcpdump and snoop. With these programs, a privileged user can see the headers and data of the TCP/IP packet.

If data travel on a closed network owned by NASA, then the problem becomes bounded and much simpler. NASA has the right to determine which users are authorized to have access to the network. However, even in this case, it can be difficult to tell who is snooping on the network, since these individuals have become very sophisticated.

Encryption can solve this problem of eavesdropping. The IETF has defined a standard called IPSec, which can be used as part of a VPN. IPSec is designed to work with IP packets and has been tested over terrestrial networks, so applying IPSec to an IP-based mission should be fairly straightforward. The packet could be encrypted on the satellite, and then regardless of the networks that the packet had to traverse, the encryption would be maintained until the end user received the data. When a packet was encrypted on the satellite, the packet would still contain an unencrypted IP header (or a subset of the IP header) that would allow networks and routers to move the packet along to its destination machine. The rest of the packet would be encrypted and would only be deencrypted by the destination.

There are other encryption algorithms, such as Secure Sockets Layer (SSL), Transport Layer Security (TLS), or High-Assurance Internet Protocol Encryption (HAIPE) that were not considered for this report. Both SSL and TLS provide encryption for the payload data only and do not encrypt the headers of the IP packet or specify how the headers should be encrypted. These schemes do not make the packet as secure as IPSec. The Department of Defense is currently working on a VPN solution using HAIPE as the encryption mechanism, but since HAIPE is still classified, it will, most likely, be used only for combined missions with the Department of Defense (ref. 9).

The downside to encryption is that it is very resource intensive. It can take a significant amount of processing power to apply some types of encryption schemes. The project will have to determine whether the satellite has enough extra processing power to perform the encryption before transmitting the packet. Encryption could impact the design of the spacecraft.

3.7 Gateways

Gateways can be placed in the communications path to enhance the protocols between the senders and receivers. Gateways intercept and convert the packet data or add options to the headers before they are transmitted over the unreliable link. A gateway allows the sending node to use existing applications to communicate over an unreliable link without modifying or rewriting the application. The sending node might not have the ability to modify the source code or it may be a commercial application and the source code may be proprietary. Conceptually, a gateway works as shown in figure 4.

The figure shows a bidirectional communication link between two wired networks that have an unreliable link in the path (i.e., the satellite). The communications path is passed through a gateway that will either convert the protocol or add options to the TCP/IP headers so that it will tolerate the properties of the unreliable link and the protocols will be enhanced.

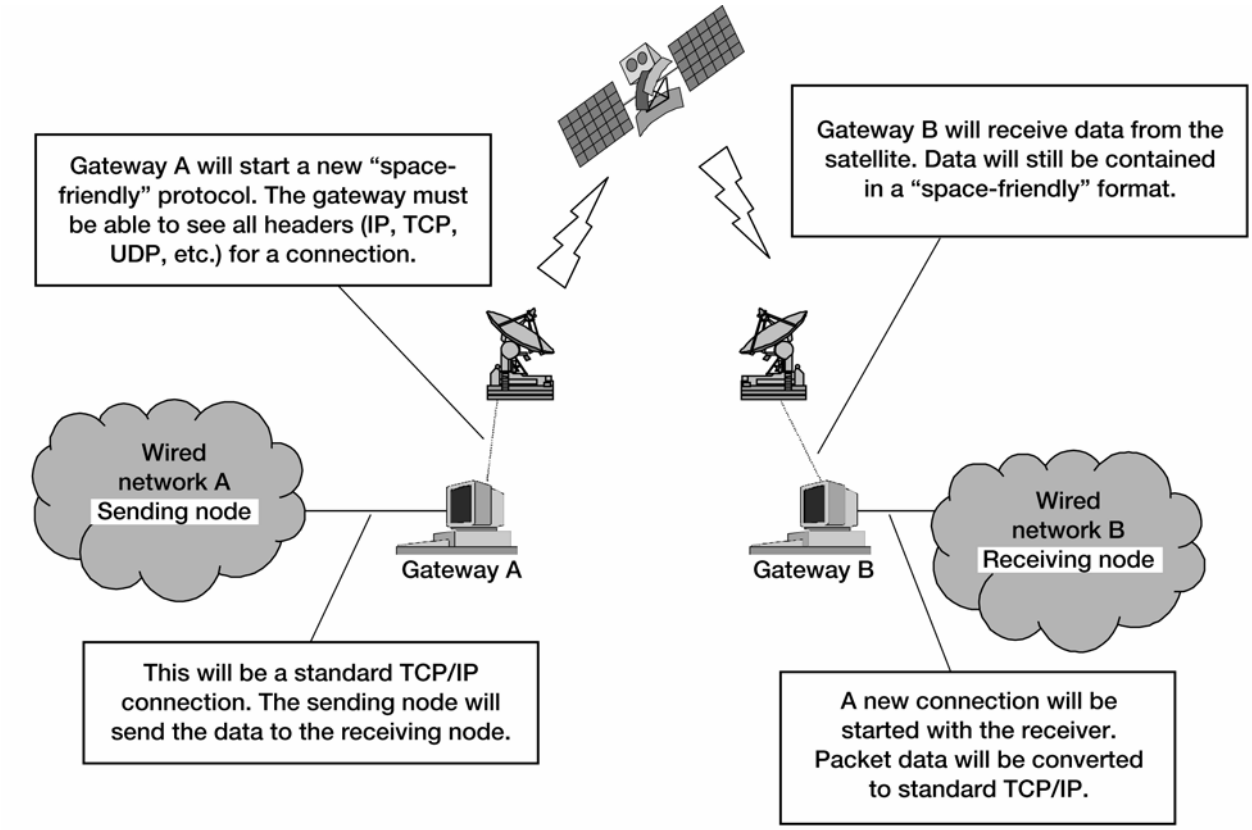

Figure 4.—Bidirectional communications link between two wired networks with gateways.

As shown in figure 4, the gateway's configuration will work as follows:

- A sending node on wired network A will send a packet of data to a receiving node on wired network B. The routing table indicates that any packet destined for the receiving node must pass through the gateway, so the packet is sent to gateway A.
- The gateway will receive the packet and start a separate connection with the receiver to transmit the packet. Note that the connection coming into the gateway and the connection leaving the gateway will be different connections. The gateway will start its own connection, since it will need different properties or protocols to communicate with the satellite. The data packet leaving the gateway will have either been converted to the new protocol or will have appended options to allow it to operate more efficiently over the unreliable link.
- The packet will then be transmitted from the ground terminal gateway to the satellite. Providing that the satellite is just a communications point, it will send the packet back to the appropriate ground terminal. In this case, it is gateway B.
- Gateway B will receive the packet and either remove the options from the TCP/IP header or convert the protocols from those used on the unreliable link to TCP/IP. Again, the connection from the satellite to the receiving node in wired network B will be intercepted by the gateway and the gateway will start a new TCP/IP connection between the gateway and wired network B. These two connections—between the satellite and the gateway and between the gateway and wired network B—will be different and could be running different protocols.
- The gateway will then forward the packet to the receiving node in wired network B.

Figure 4 has been simplified; a real communications path could contain encryptors, firewalls, routers, and other networking equipment. The figure assumes that these are contained in the wired network A and

B clouds. In addition, the path from wired network B to wired network A will be very similar to the scenario described.

3.8 Encryption and Gateways

Encryption, as discussed in section 3.6 is very powerful, and it allows data to traverse a satellite link and arrive at the destination without anyone being able to determine the contents of the packet. This works under the new design where the entire mission is IP-based. For the most confidentiality, the data should be encrypted at the sending node and transmitted to the receiving node; it is the receiver's responsibility to deencrypt the packet data.

As mentioned in the previous section, the gateway must break the incoming connection and start a new connection with the next element (e.g., the receiving node) in the path. However, using IPSec, the gateway does not have access to the transport header information, since not only are the payload data encrypted, but also the transport header and part of the IP header. In this scenario, the gateway would not know to which port numbers on the receiving machine the packet is destined or even if the transport header is TCP or User Datagram Protocol (UDP). The gateway would need to deencrypt the data packet to get access to the full headers to make the next connection, but the gateway may not know how to deencrypt the packet. Even if it could deencrypt the packet, it would leave itself vulnerable to a man-in-the-middle attack (i.e., a break-in at the gateway). With the packet deencrypted, both the data and the headers will be "in the clear" at the gateway. Therefore, since the gateway would need access to the full header to optimize the connection, gateways are not recommended communication paths that have true end-to-end encryption from the sender to the receiver.

4.0 Results and Discussion

The focus of this report is on exploring and developing an end-to-end IP-compliant communications design. NASA plans to develop more complex systems from constellations to space networks to sensor webs. These systems will be responsible for not only communicating among each other but also with ground terminals. With these systems, autonomous, interoperable, and distributed operations become more important. So a communication mechanism is needed that can be used seamlessly between the space and ground systems. The most probable candidate would be the TCP/IP suite of protocols, given their wide use on terrestrial systems.

This report divides a generic satellite mission into two pieces. First, the design for the satellite bus is described as shown in figure 2. The goal was to develop a bus that would be IP-based from the source (data collection) through the transmission to the ground. Second, the communication links between the satellite and the ground entities are described in figure 3. For each of these links, a description is given of the link and the type (or types) of protocols that are recommended.

The satellite bus is the central component to the design, as shown in figure 2. The design goal was to use standard terrestrial networking designs and components but to ensure that security was given a main priority. All data that are uplinked or downlinked to or from the satellite will have to pass validity checks through a firewall. The communications stream will be protected via a VPN, which can ensure authentication and access control. One advantage to this design is that it can be configured to fit the mission requirements. For example, researchers will be able to determine the number of subnets for the bus design and the modules contained on those subnets. Each subnet should be designed to the mission goals. In our design, the command and control instruments are on one subnet and the science instruments are on another. If desired, the science instruments could be divided among more than one subnet if there are security, data volume, or other concerns. The router will be responsible for transmitting the data packets onto the correct subnet.

Figure 3 shows the communication links between the satellite and major entities, such as the users, the data processing system, and the field sites. The links provide communications for uplinking or downlinking commands and data. The space-based, or radiofrequency links, are divided into two categories: reliable and unreliable protocol links. Reliable protocol links are either links where data must be transferred reliably, securely, and correctly the first time (e.g., satellite commands) or those where file sizes are small and the transfer can handle the overhead of the built-in reliability. Reliable links should use TCP/IP for data transmissions. Unreliable links are those where it does not matter whether or not the data get to the destination correctly the first time. Even though these links are called unreliable, the proposed protocols do have built-in error-correction schemes to ensure that all the data eventually arrive at the destination. Unreliable link data can be characterized as large files that do not contain critical data. Proposed protocols for these data would be TCP with SNACKS, MDP, or SCTP.

This report shows that an IP-based mission is plausible, but there is still more research and engineering work to accomplish. Mission architectures must be developed and integrated into the mission designs. Either the network components used on the bus must be radiation hardness tested, or space-based versions of these components must be developed. In addition, the missions must be analyzed and corrected for security problems. With these steps, NASA can plan for future IP-based missions.

5.0 Conclusions

For many scientists, the most important part of a satellite mission is not the communications infrastructure, but whether or not the data can be collected and transmitted to the ground successfully for analysis. As long as this can be done, how the data get to the ground is not a major concern. Although this is a limited view, it is prevalent among the science communities. This report shows that an IP mission is conceptually possible using today's technologies and may provide cost savings to missions, if implemented correctly. Most importantly, this infrastructure has the potential of being reused across missions.

NASA has the ability to create an IP-compatible satellite bus that can collect data in an IP-based protocol. The satellite can be designed to operate in a safe and secure environment by having all data pass through a firewall and ensuring that the packets are validated. In addition, all ground-to-satellite communication will be via a VPN. With a VPN, the data can be securely transmitted to the ground, so that intruders cannot tamper with them. Once the packets reach the terrestrial environment, public networks can transport them to the worldwide science community.

However, mission and/or system requirements will determine the extent, if any, that standard protocols and interfaces, such as IP and Ethernet, can be used on a mission. The particular requirements of the mission must be examined to determine how many benefits exist. Although the terrestrial market has seen tremendous cost savings by utilizing standard interfaces and protocols, such savings might not be realized in the space market. Therefore, this report provides mission designers with a foundation. They can utilize this material, but they will need to determine how an end-to-end IP design will meet their requirements.

Appendix A

Differences Between Wired and Wireless Networks

In appendixes A and B, we introduce the Internet Protocols and discuss the ones that have been optimized for use over unreliable links. These protocols are modified, since wireless systems have different characteristics than wired systems (ref. 10). Wireless systems not only include space-based systems, but also terrestrial systems, such as cellular phones and 802.11 wireless networks. This section lists the differences between wired and wireless systems:

- *Bit error rate.* Bit error rates can be higher in wireless communications, ranging from 10^{-5} to 10^{-12} depending on the link quality.
- *Continuity of connectivity.* With wireless communications, links are unreliable and link outages can occur frequently.
- *Forward and reverse links.* In wired communications, it is reasonable to assume that bandwidths of bidirectional links do not differ significantly. But in space communications, this is not a reasonable assumption. The range can be 10:1 to 2000:1.
- *Central processing unit capacity and memory availability.* Processors used on satellites usually lag the terrestrial industry, so they have limited processing power and memory.
- *Congestion versus corruption.* In wired networks, losses are assumed to be due to congestion, since congestion, rather than corruption, has been the dominant problem. By default, terrestrial TCP/IP stacks assume that losses are due to congestion. However, on a wireless network, corruption, due to delays and bit error rates, is as much as, or more of, a problem than congestion is.
- *Link utilization.* The contact time that a ground station has to communicate with a satellite may be limited. The link must be fully utilized to get as much data to the receiver as possible. In space communications, disabling slow-start or congestion control algorithms would allow the greatest bandwidth utilization.

Appendix B

TCP/IP Suite of Protocols

B.1 IP, UDP, and TCP

For computers to transmit data over the Internet, they need a standard way to communicate. They must understand the format and structure of the data and agree upon a set of rules for passing the data. This standard method of communicating is called a protocol, and the most prevalent protocols on the Internet are TCP and UDP. Both TCP and UDP use IP as the lower-layer protocol, hence the terminology TCP/IP Suite (ref. 11).[3] Networking protocols are usually developed in layers and follow a standard model. The four layers in the TCP/IP protocol suite are shown in table I.

TABLE I.—TCP/IP SUITE OF PROTOCOLS
[HTTP, Hypertext Transfer Protocol; ICMP, Internet Control Message Protocol; IGMP, Internet Group Management Protocol; NIC, network interface card.]

Application	Telnet, FTP, HTTP
Transport	TCP, UDP, SCTP
Network	IP, ICMP, IGMP
Link	NIC, Ethernet, modem

Data start at the application level and progress downward through the model to the link layer where they pass through the NIC or modem and out onto the channel. Data files are broken into chunks called packets. As the data pass through the layers of the model, headers, and in some cases trailers, are attached. This section briefly discusses IP, TCP, and UDP protocols and the differences among them.

B.2 Internet Protocol (IP)

The main goal of the Internet Protocol (IPv4)[4] is to enable the routing of data from a sender to a receiver over a mesh-type network. IP provides the lower-layer protocol for both TCP and UDP. It appends a 20-byte header to either the TCP or UDP header as the packet passes through the network layer. By itself, IP is connectionless and unreliable. “Connectionless” means that it does not maintain any type of state between the sender and receiver and that packets can be delivered out of order. IP is also unreliable: the sender has no guarantee that any packet is delivered correctly (i.e., IP has no acknowledgment system). If a packet is received in error according to a checksum, the packet is dropped and no information is returned to the sender. Both reliability and state information must be added in either another protocol layer (e.g., TCP) or by the application.

IP provides three important features:

(1) It provides the source and destination addresses: it knows how to deliver the packets to the appropriate destination; and once the destination gets the packet, it knows the return (sender) address.

(2) It provides the type of transport protocol so that the IP layer knows if this is a TCP or UDP packet.

(3) It provides a 16-bit checksum so that the packet can be validated against errors that occurred during transmission. The checksum is only calculated on the IP header of the packet and not on the payload.

[3]The TCP/IP Suite is a suite of actual protocols. This report will focus on TCP, UDP, and IP.
[4]In this document, IP refers to IP version 4

B.3 Transmission Control Protocol (TCP)

The next layer in the TCP/IP model (see table I) is the transport layer that contains TCP (refs. 12 and 13). TCP is a reliable and connection-oriented protocol. Reliable implies that every packet transmitted by the sender via TCP has an acknowledgment returned by the destination. If the destination does not acknowledge packets within a certain amount of time, then the packets are re-sent. Since TCP is connection-oriented, the sender must establish a connection via a three-way handshake with the destination before transferring data packets.

Additional advantages of TCP are slow start and congestion control. Slow start is a technique used by TCP to ascertain the available end-to-end bandwidth. If the acknowledgments are being returned successfully, TCP will exponentially increase its send rate. The connection will continue to increase the rate if congestion is not detected (e.g., acknowledgments are not being received by the sender). If congestion is encountered, the congestion control algorithms will be activated. The data rate will be halved and linearly increased to conservatively ascertain the available bandwidth.

The TCP header contains the following three types of information: (1) the port numbers of the sender and receiver that indicate on which port the destination is listening for a connection and from which port the client is sending data, (2) sequence and acknowledgment numbers to implement reliability, and (3) TCP checksums used to validate the data packet.

B.4 User Datagram Protocol (UDP)

In addition to TCP, UDP resides in the transport layer (see table I). Similar to IP, UDP is connectionless and unreliable. As mentioned previously, unreliable means that the sender has no guarantee that the recipient received the packet (i.e., no confirmation). “Connectionless” means that the sender and receiver do not maintain the state of the connection and that each packet is sent as a new packet to the destination. The checksum can be used to check the integrity of packets that reach the destination. In addition, there is no packet reordering or congestion control in UDP.

The UDP header attaches two pieces of information to the IP header: the port numbers and a checksum. The port number of the sender indicates the port that the client is sending data from, and the port number of the receiver indicates the destination port that is listening for a connection. The UDP header also contains a 16-bit checksum to provide data integrity, but since the application has the option of not generating a checksum, it is not always applied to that packet (ref. 14).

Appendix C

Space Communications Protocol

C.1 Space-Based Protocols

To improve the performance of TCP/IP on unreliable links, projects have attempted to modify the terrestrial protocols or invent new protocols for space-based applications. In this section, four protocols are presented: the Space Communications Protocol Standard (SCPS, ref. 15), XIP-Link (ref. 16), SkipWare (ref. 17), and MDP.

C.2 Space Communications Protocol Standard (SCPS)

SCPS (http://www.scps.org), developed jointly by NASA, the Jet Propulsion Laboratory, and the Department of Defense, addresses the issues of data communications over wireless networks. SCPS is not a unique, different, or proprietary protocol, but one that is based on the TCP/IP suite with options to increase the performance of unreliable communications links, such as those between ground stations and satellites. SCPS contains four layers:

- *SCPS–FP*. This layer provides the standard FTP application and operates at the application level of the TCP/IP model.
- *SCPS–TP*. This layer provides the transport layer and is optimized to provide reliable data communications over a network containing one or more unreliable links.
- *SCPS–NP*. This layer supports both connectionless and connection-oriented routing of packets over space or wireless data links. NP is based at the network layer.
- *SCPS–SP*. This layer provides end-to-end security and integrity of messages. This does not match any of the layers defined in the Internet five-layer or Open Systems Interconnection (OSI) seven-layer model.

The SCPS project was tasked with developing specifications that define revisions to IP to permit them to operate over unreliable, highly latent, asymmetric bandwidth links. The users of SCPS not only use the layers (e.g., SCPS–TP and SCPS–NP) of SCPS as fully defined, but they can also use parts of these layers to meet mission requirements. Two important modifications are TCP with SNACKS and user-selectable congestion control. TCP with SNACKS allows users to have built-in reliability in an unreliable protocol by having the destination acknowledge only those packets that did not transfer successfully and having the sender retransmit only those packets. In addition, users can decide the type of congestion control—Van Jacobson (ref. 18), Pure Rate, or TCP-Vegas (ref. 19)—to use in the data transfer. CCSDS has adopted SCPS as part of their standard to provide an IP base for CCSDS.

C.3 Xip-Link

Xip-Link, developed by Xiphos Technologies, is a commercial implementation of the SCPS–TP layer. Xiphos developed Xip-Link on the basis of the specifications derived from the SCPS project; Xip-Link provides an in-kernel implementation of SCPS–TP.

C.4 SkipWare

SkipWare, developed by Global Sciences and Technologies, provides another commercial implementation of the SCPS standards. Developed on the specification from the SCPS project, SkipWare provides a gateway implementation so that legacy applications can communicate over the unreliable link.

C.5 Multicast Dissemination Protocol (MDP)

MDP (ref. 20) is being developed at the Naval Research Lab; the original code was developed around 1995. The goal of the MDP project is to design a reliable, scalable, and efficient protocol that uses multicasting for data delivery. The protocol is optimized to work on heterogeneous networks and is intended for both wired and wireless networks.

The basic idea is that a satellite can simultaneously broadcast to multiple ground stations using multicast addressing based on UDP. Since MDP uses UDP, which by nature is unreliable, the Naval Research Lab has enhanced the protocol to add reliability. When this approach is used, MDP can take advantage of UDP by reducing the handshaking and overhead of acknowledging all of the data transferred. To ensure reliability, MDP uses a model called Selective Negative Acknowledgments (NACKs) by which the sender sends data to multiple receivers and the receiver acknowledges only those packets that they did not receive. The receivers can multicast the NACKs (to the source and other receivers) or can unicast them (directly back to the sender).

References

1. Postel, J.; and Reynolds, J.K.: File Transfer Protocol. RFC 959, 1985.
2. Fielding, R., et al.: Hypertext Transfer Protocol. RFC 2616, 1999.
3. Connolly, D.; and Masinter, L.: The ‘text/html’ Media Type. RFC 2854, 2000.
4. Levi, D.; Meyer, P.; and Stewart, B.: Simple Network Management Protocol (SNMP) Applications. RFC 3413, 2002.
5. Consultative Committee for Space Data Systems. http://www.ccsds.org/ Accessed Mar. 20, 2003.
6. Kent, S.; and Atkinson, R.: Security Architecture for the Internet Protocol. RFC 2401, 1998.
7. Stone, J.; and Partridge, C.: When the CRC and TCP Checksum Disagree. Comput. Commun. Rev., vol. 30, no. 4, 2000, pp. 309–319.
8. Virtual Private Networks (VPNs). International Engineering Consortium, IEC Web ProForum Tutorials CD–ROM, vol. 9, 2003. http://www.iec.org Accessed Mar. 20, 2003.
9. SafeNet. http://www.safenet-inc.com Accessed Mar. 20, 2003.
10. Allman, M.; Glover, D.; and Sanchez, L.: Enhancing TCP Over Satellite Channels Using Standard Mechanisms. RFC 2488, The Internet Society, 1999.
11. Stevens, W. Richard: TCP/IP Illustrated. Addison-Wesley Publishing Co., Reading, MA, 1996.
12. Postel, J.: Transmission Control Protocol, DARPA Internet Program Protocol. RFC 793, 1981.
13. Postel, J.: Internet Protocol, DARPA Internet Program Protocol Specification. RFC 791, 1981.
14. Braden, R.: Requirements for Internet Hosts—Communication Layers. RFC 1122, 1989.
15. Space Communications Protocol Standards. http://www.scps.org Accessed Mar. 20, 2003.
16. Xiphos Technologies. http://www.xiphos.com Accessed Mar. 20, 2003.
17. Skipware. http://skipware.com/ Accessed Mar. 20, 2003.
18. Jacobson, Van: Congestion Avoidance and Control. Comput. Commun. Rev., vol. 18, no. 4, 1988.
19. Brakmo, L.S.; O’Malley, S.W.; and Peterson, L.L.: TCP Vegas: New Techniques for Congestion Detection and Avoidance. Comput. Commun. Rev., vol. 24, no. 4, 1994, p. 24.
20. Adamson, Brian; and Macker, Joe: The Multicast Dissemination Protocol (MDP). Naval Research Laboratory MDP Protocol Specification Version 1.6, 1999. http://manimac.itd.nrl.navy.mil/MDP/DraftMdpSpec-1.6.txt Accessed Mar. 25, 2003.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	February 2004	Technical Memorandum

4. TITLE AND SUBTITLE

Low-Earth-Orbit Satellite Internet Protocol Communications Concept and Design

5. FUNDING NUMBERS

WBS–22–258–90–05

6. AUTHOR(S)

Richard A. Slywczak

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

National Aeronautics and Space Administration
John H. Glenn Research Center at Lewis Field
Cleveland, Ohio 44135–3191

8. PERFORMING ORGANIZATION REPORT NUMBER

E–13863

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

National Aeronautics and Space Administration
Washington, DC 20546–0001

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

NASA TM—2004-212299

11. SUPPLEMENTARY NOTES

Responsible person, Richard A. Slywczak, organization code 5610, 216–433–3493.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Unclassified - Unlimited
Subject Category: 17 Distribution: Nonstandard

Available electronically at http://gltrs.grc.nasa.gov

This publication is available from the NASA Center for AeroSpace Information, 301–621–0390.

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 words)***

This report presents a design concept for a low-Earth-orbit end-to-end Internet-Protocol- (IP-) based mission. The goal is to maintain an up-to-date communications infrastructure that makes communications seamless with the protocols used in terrestrial computing. It is based on the premise that the use of IPs will permit greater interoperability while also reducing costs and providing users the ability to retrieve data directly from the satellite. However, implementing an IP-based solution also has a number of challenges, since wireless communications have different characteristics than wired communications. This report outlines the design of a low-Earth-orbit end-to-end IP-based mission; the ideas and concepts of "Space Internet" architectures and networks are beyond the scope of this document. The findings of this report show that an IP-based mission is plausible and would provide benefits to the user community, but the outstanding issues must be resolved before a design can be implemented.

14. SUBJECT TERMS

Satellite; Internet protocols; Communications

15. NUMBER OF PAGES

28

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	