IPN Research Group
Internet Draft
May 2001
Expires November 2001

V. Cerf
Worldcom/Jet Propulsion Laboratory
S. Burleigh
A. Hooke
L. Torgerson
NASA/Jet Propulsion Laboratory
R. Durst
K. Scott
The MITRE Corporation
E. Travis
Global Science and Technology
H. Weiss
SPARTA, Inc.

Interplanetary Internet (IPN): Architectural Definition

draft-irtf-ipnrg-arch-00.txt

Status of this Memo

Abstract

This document describes the Interplanetary Internet: a communication system to provide Internet-like services across interplanetary distances in support of deep space exploration. Our approach, which we refer to as bundling, builds a store-and-forward overlay network above the transport layers of underlying networks. Bundling uses many of the techniques of electronic mail, but is directed toward interprocess communication, and is designed to operate in environments that have very long speed-of-light delays. We partition the Interplanetary Internet into IPN Regions, and discuss the implications that this has on naming and routing. We discuss the way that bundling establishes dialogs across intermittently connected internets, and go on to discuss the types of bundle nodes that exist

in the interplanetary internet, followed by a discussion of security in the IPN, a discussion of the IPN backbone network, and a discussion of remote deployed internets.

Table of Contents

Desiderata of Interplanetary Internetworking

Go thoughtfully in the knowledge that all interplanetary communication derives from the modulation of radiated energy, and sometimes a planet will be between the source and the destination. Therefore rely not on end-to-end connectivity at any time, for the universe does not work that way.

Neither rely on ample bandwidth, for power is scarce out there and the bit error rates are high. Know too that signal strength drops off by the square of the distance, and there is a lot of distance.

Consider the preciousness of interplanetary communication links, and restrict access to them with all your heart. Protect also the confidentiality of application data or risk losing your customers.

Remember always that launch mass costs money. Think not, then, that you may require all the universe to adopt at once the newest technologies. Be backward compatible.

Never confuse patience with inaction. By waiting for acknowledgement to one message before sending the next, you squander tracking pass time that will never come to you again in this life. Send as much as you can, as early as you can, and meanwhile confidently await responses for as long as they may take to find their way to you.

Therefore be at peace with physics, and expect not to manage the network in closed control loops -- neither in the limiting of congestion nor in the negotiation of connection parameters nor even in on-demand access to transmission bands. Each node must make its own operating choices in its own understanding, for all the others are too far away to ask. Truly the solar system is a large place and each one of us is on his or her own. Deal with it.

S. Burleigh

Acknowledgments

Robert Braden, of USC ISI, Deborah Estrin, of UCLA, and Craig Partridge, of BBN all contributed useful thoughts and criticisms to this document.

This work was performed under DOD Contract DAA-B07-00-CC201, DARPA AO H912; JPL Task Plan No. 80-5045, DARPA AO H870; and NASA Contract NAS7-1407.

Foreward

This document presents the current state of our team's efforts to define an end-to-end architecture for the Interplanetary Internet. This is a "living" document, and is frequently updated. In this version of the document, we introduce a new construct, a tuple consisting of a topologically significant routing handle and the administrative name. In this model, the routing handle identifies an "IPN Region," an area of the Interplanetary Internet in which A) the administrative name is resolvable, and in which B) a route can be formed from anywhere within the region to the address returned when the administrative name is resolved.

On the presumption that the exploration of space will eventually lead to the need for communication among planets, satellites, asteroids, robotic spacecraft and crewed vehicles, our project has a heavy focus on advocating the development of a stable interplanetary backbone network. We have concluded that simply extending the Internet suite to operate end-to-end over interplanetary distances is not feasible. Rather, we envision a "network of internets" _ ordinary internets that are interconnected by a system of gateways that cooperate to form the stable backbone across interplanetary space. Each internet's protocols are terminated at its local gateway, which then uses a specialized long-haul transport protocol to communicate with peer gateways. An end-to-end "bundle" protocol will operate above the transport layer to carry necessary information from one internet to another. Bundling will, to the extent possible, remove any "chattiness" from the local protocols, forming atomic units that will be shipped across the backbone. Bundles may be protected from unauthorized access and unauthorized modification. The IPN will have a global namespace that is broken into a number of name-to-address binding regions, referred to as IPN regions. Names carried in the bundles consist of a tuple, one element identifying the destination IPN region and used for routing and a second element that carries the administrative name of the destination in the namespace that is relevant within the destination IPN region. The administrative name will be bound to an address that is routable within the destination IPN region. Finally, strong authentication and strict access controls at several levels will protect the IPN from tampering.

In summary, the best way to envision the fundamental architecture of the Interplanetary Internet is to picture a network of internets. Ordinary internets (many being wireless in nature) are placed on the surface of moons and planets as well as in free-flying spacecraft. These remotely deployed internets run ordinary Internet protocols. A system of Interplanetary Gateways connected by deep-space transmission links form a backbone communication infrastructure that

provides connectivity for each of the deployed internets. New long-haul protocols, some confined to the backbone network and some operating end-to-end, allow the deployed internets to communicate with each other.

Executive Summary

This document describes the Interplanetary Internet: a communication system to provide Internet-like services across interplanetary distances in support of deep space exploration. The communications environment is characterized by high bandwidth-delay products resulting from very long signal propagation delays, intermittent connectivity that results in long periods of network partitioning, and discontinuities in the capabilities of adjacent networks. Many of these characteristics are similar to those facing emerging communication services in the terrestrial Internet. For example, terabit networks exhibit very high bandwidth-delay products, mobility can result in partitioning of both nodes and subnetworks, and interconnecting different physical layer technologies can result in "impedance mismatches." It is possible to build an end-to-end solution to address any of these, but difficult to build one that is adequate for all of them simultaneously.

The long bandwidth-delay products shared by the IPN and very high-speed terrestrial networks argue in favor of non-chatty communication protocols. In the IPN, the long delays mean that protocols that use many round-trips to accomplish some task pay a significant time penalty. In terrestrial terabit networks where switches can forward data very fast but take a (relatively) long time to reconfigure, the penalty is one of lost efficiency in the use of the resources. Both environments benefit from protocols that pack as much as possible into each transmission and minimize the number of round-trips needed. Thus a file transfer protocol that can place the entire file and associated control information together in a single atomic transaction completes faster within the IPN and makes more efficient use of terrestrial high-speed networks.

Most of the problems cited above have existed and been considered, albeit separately, during the evolution of what is now the Internet. Many of the solutions, however, represent branches of the Internet's evolutionary tree that have withered and died as a result of the availability of infrastructure to mitigate environmental differences. This effective homogeneity is diminishing, however, with the rapid deployment of new technologies with fundamentally different characteristics, such as wireless communication and Dense Wavelength Division Multiplexing (DWDM) networks. One solution, electronic mail, provides a means to span very different networks that are not necessarily always connected. The electronic mail approach has a number of attractions. First, there is no expectation of continuous or instantaneous connectivity. Second, electronic mail embodies the concept of indirection as a means of providing store-and-forward traversal of different, sometimes-disconnected networks. Finally,

the electronic mail concept is generally considered to be a non-interactive communications mechanism, potentially well suited to long delay environments. The electronic mail approach has limitations, though. Without and end-to-end retransmission mechanism, electronic mail does not provide true end-to-end reliability. Additionally,

electronic mail is oriented toward human use rather than interprocess communication. Further, the protocol that typically provides electronic mail services in the Internet, SMTP, is highly interactive in its control traffic, even though the electronic mail concept is only minimally interactive.

Our approach, which we refer to as bundling, builds a store-and-forward overlay network above the transport layers of underlying networks. Thus two nodes that are adjacent in bundle space may be many hops apart in the context of the underlying network topology. We see the bundle layer as a second "thin waist of the hourglass" that allows applications built on top of it to communicate across discontinuities in connectivity, and to communicate efficiently over a multitude of underlying transport technologies. This discontinuity in connectivity may result from fluctuations in link availability or from artificial discontinuities, such as are imposed by firewalls. For efficient communication, the bundle layer attempts to minimize interactivity of its control traffic, and expects applications to do likewise. The bundle layer also provides a level of indirection between applications and the specific services of the underlying network protocols. Bundle applications can specify requested "handling instructions," such as reliability and quality of service requests that are mapped into the most appropriate mechanisms available in the underlying networks.

Bundling uses many of the techniques of electronic mail, but is directed toward interprocess communication. Bundle nodes use the capabilities of the underlying networks, including transport layer retransmission protocols, to effect the transfer of bundles between bundle nodes. Optional end-to-end reliability at the bundle layer facilitates end-to-end reliability at the application layer. In addition, the bundle layer allows any bundle node in the path to take custody of a bundle. When custody is transferred, the receiving bundle node assumes responsibility for delivering the bundle according to its handling instructions, and the previous bundle custodian is allowed to recover its storage resources. The bundle protocol is designed to function over simplex and half-duplex links, and custody transfers may occur between non-adjacent bundle nodes. Finally, because the bundle layer operates over networks that are often disconnected, the bundle-layer reliability mechanisms adapt the operation of timers to accommodate this episodic connectivity.

To illustrate how bundling allows connectivity across intermittently connected networks, one could envision a "strobe light" that illuminates the portions of the network's topology that are connected at a given time. Lit portions of the network are available for bundle forwarding. If one imagines a high-persistence CRT capturing

an ordered sequence of illuminations that proceed from source to destination, one can envision the available routes for bundle forwarding. This environment, therefore, requires a mechanism to route based on both current and expected connectivity. This routing mechanism exploits predictability, such as that provided by orbital

mechanics or by scheduled network events (possibly including rate changes, such as "5c Sundays"). It is important to note that this routing mechanism may choose to defer communication even though a current path toward the destination exists, if a substantially more attractive path is expected to become available.

We believe that the bundle layer functionality has utility within the context of the terrestrial Internet as well. The Internet is evolving to encompass very different networking technologies, architectures, and applications. Some of these, such as firewalls, result in logical partitioning of the Internet, while others, such as extreme rate mismatches, may result in inefficient use of network resources. The bundle layer extends the Internet architecture to provide consistent end-to-end communication in the current and emerging partitioned environments, facilitating the deployment of new applications that can operate reliably while allowing networks to operate efficiently.

Readers may find more about the project at http://www.ipnsig.org/, the Interplanetary Internet Special Interest Group of the Internet Society (http://www.isoc.org).

1. Introduction

The exploration of space began with naked-eye observations of the stars and moon. In more recent centuries, this exploration was aided by new Earth-based technologies such as optical and radio telescopes. Even more recently, we have placed observing resources into near Earth orbit, such as the Hubble Space Telescope. We have also sent robotic missions to other parts of the Solar system for a closer look.

It is the latter activity -- the current exploration of the Solar System by robotic means and possibly later by missions crewed by people -- that motivates our interest in an Interplanetary Internet. The new technology of the terrestrial Internet needs to be extended into space. We believe that the creation and adoption of Internet-friendly standards for space communication will enhance our ability to build a common interplanetary communication infrastructure. We think this infrastructure will be needed to support he expansion of human intelligence throughout the Solar System. The current terrestrial Internet and its technology provide a robust basis to support these missions in an efficient and scalable manner.

Although many missions during the last 40 years of space exploration have by necessity provided a substantial portion of their own communication resources, a significant amount of shared infrastructure is already in place. For example, the multi-mission Deep Space Network (DSN) provides a complex of large-diameter (up to 70-meter) tracking and data acquisition dishes at three points on the Earth's surface, each about 120 degrees from each other. Similarly, the Tracking and Data Relay Satellite System (TDRSS) and a global network of small ground tracking stations are used to relay data to and from many near-Earth missions.

In terms of the communications protocols that support space exploration, the Consultative Committee for Space Data Systems (CCSDS) has for almost twenty years been developing internationally agreed standards for the physical and link layers that interconnect remote spacecraft with their ground control systems. NASA, through CCSDS, has also been working since 1993 on general application of terrestrial Internet or Internet-like protocols for space data use. Such standardization opens up the possibility of re-purposing and re-using existing and planned communication facilities for multiple subsequent missions.

Early in 1998, it became apparent to our team that the space communications research community and the Internet research and

development community were pursuing technology paths that can potentially lead to a kind of convergence. A few members of the Internet community began thinking about adaptation of the terrestrial Internet to deep space communications. The space communications

research community was already trying out variants of the Internet's TCP/IP protocol suite to support space-based applications. Mutual recognition led to the formation of a program of work that was aimed at extending the notion of Internet to interplanetary scale. The heretofore-independent "rivers" of evolving space technology and Internet technology are converging in this program.

To realize this convergence, the effort to define the IPN architecture is being undertaken at the Jet Propulsion Laboratory (JPL) that is operated by the California Institute of Technology (Caltech) for the US National Aeronautics and Space Administration (NASA). Partial funding for the effort comes from the US Defense Advanced Research Projects Agency (DARPA) that sponsored the original Internet design work starting in 1973 and its predecessors, such as the ARPANET, in 1969. NASA supplies in-kind support and staffing through its standardization program with CCSDS as well as program involvement from the Mars exploration enterprise.

An Interplanetary Internet Research Group (IPNRG) has been formed under the auspices of the Internet Research Task Force of the Internet Society and an Interplanetary Internet Special Interest Group has also been created as a means of keeping the public informed as to progress.

Early protocol design phases are underway now and prototype testing of candidate designs is anticipated within a year. Demonstration of these protocols in terrestrial environments will likely occur during 2001 and plans for their use in interplanetary contexts are under consideration as part of the NASA Mars mission in the years beyond 2003. The Mars exploration program is particularly interesting as a "reference model" for our work, since it includes a "Mars Network" project that hopes to deploy a series of remote communications satellites into orbit around the planet. These satellites will be dedicated to servicing the local communication and positioning requirements of the other in-orbit and surface observation and exploration missions that are in the vicinity of the planet. By 2010, as many as seven communications and navigation satellites could be in orbit around Mars, most in low Mars orbit (LMO) but with perhaps at least one in an "areo" synchronous orbit that is analogous to a geosynchronous orbit around the Earth.

1.1. Preliminary Considerations

The remarkable success and growth of the Earth-bound TCP/IP protocols of the Internet illustrate the power of communication standardization. The simplicity of the Internet architecture, with its layered structure, contributes to its ability to adapt to almost

any underlying communication capability. As with the terrestrial Internet, the ultimate test of the IPN technology is whether it successfully supports commercial applications that have a space component. We expect that space will eventually be commercialized - not only for communication services, but also for mining the

asteroids, for the operation of space-based hotels, for manufacturing and medical treatments, and for general tourism. While such developments may still lie decades in the future, the potential investment and benefits can be appreciated as we contemplate the explosion of new markets associated with the commercialization of the Internet that began only ten years ago, in 1990. We will therefore architect the Interplanetary Internet in anticipation of possibly rapid commercialization.

In terms of technologies, the current Internet capabilities work well on Earth where the propagation delay of light-speed signals is short. The packets exchanged according to the TCP protocol reach their destination in fractions of a second, for the most part. The TCP/IP protocols (a system of over 150 related communication standards), are therefore expected to work just as well on the surface of other planets or moons, on space craft and orbiting space stations, all of which involve data exchange over fairly short distances, subject to the availability of sufficient power to maintain good signal-to-noise ratios. However tempting it is to employ similar concepts in extending the Internet into deep space, there are problems - deep space communications really still are "rocket science." The distances between the planets are, well, astronomical. For example, the round-trip propagation delays - at the speed of light - between Earth and Mars range from about 8 minutes to over 40 minutes. This makes "chatty" protocols like TCP relatively unattractive because of their heavy dependence on near real-time exchanges between the communicating parties.

These large distances also impair the data rates that can be sustained because of radio signal degradation and attenuation. Moreover, the celestial mechanics of the solar system mean that the distances between the planets change with time. While these changes are essentially calculable, they still cause variations in delay, in transmission capacity and occasionally in connectivity due to occultation of satellites as they orbit a planet, or of ground-based facilities as planets rotate.

Size, weight and - most of all - power are supreme challenges for space-based communication systems, as they are for ground-based mobile systems. Launching mass into interplanetary trajectories, injecting mass into orbit, and landing mass into the gravity well of another planet is currently very expensive. Mass translates directly into the local availability of power. Efficient use of the communications channel allows more information to be carried per unit of transmitted power. But power limitations introduce asymmetries in the communication capacities available between Earth, for example, and the remote spacecraft and planets. There can be factors of ten or

more differences between the data rate that can be received on Earth and the rates of reception of off-Earth resources. It is quite common to be able to receive transmissions from Mars at 100 kilobits/second while the Mars-based systems may only receive from Earth at 1 kilobits/second.

All of these effects combine to make the design of an interplanetary backbone communication system a considerable challenge. The Deep Space Network - the current interplanetary backbone - uses its three terrestrial communication complexes to communicate with spacecraft, orbiting satellites and ground-based resources on moons or other planets of the Solar System. Because these resources must be shared among many missions, it is necessary to schedule them to be aimed in particular directions at particular times. Time synchronization is needed among the various parts of such a system. For example, a signal from Mars may take 20 minutes to reach Earth, at which time, the appropriate antenna of the Deep Space network must be aimed properly to receive the transmission, 20 minutes after it was sent. This same antenna might then have to be repositioned to send data to another spacecraft elsewhere in the Solar System, and the receiving system must be ready to receive that transmission at the right time. In some ways, this problem is somewhat like the problem of scheduling trains on railroad tracks; since multiple trains use the tracks, they must be scheduled to avoid collisions.

1.2. The IPN Operating Environment

There are a number of fundamental differences between the environments for terrestrial communications and those we envision for the IPN. These differences include delay, low and asymmetric bandwidth, intermittent connectivity, and a relatively high bit error rate. Taking these into account affects the entire model for communicating; shifting us from the "telephony" model implicit in current Internet communications to the "Postal", or "Pony Express," model. We will first therefore describe the environmental differences between terrestrial communications and the IPN and gives a brief accounting of why the standard Internet protocol for reliable transport, TCP, is unsuitable for end-to-end communications in the IPN.

The most obvious difference between communicating between points on Earth and communicating between planets is the delay. While round-trip times in the terrestrial Internet range from milliseconds to a few seconds, round-trip times to Mars range from 8 to 40 minutes, depending on the planet's position, and round-trip times between Earth and Europa run between 66 and 100 minutes. In addition to the propagation delay, communicating over interplanetary distances currently requires special equipment (large antennas, high-performance receivers, etc.). For most deep-space missions, even non-NASA ones, these are currently provided by NASA's Deep Space Network (DSN). The communication resources of the DSN are currently oversubscribed and will probably continue to be so in the future.

While studies have been done as to the feasibility of upgrading or replacing the current DSN, the number of deep space missions will probably continue to grow faster than the terrestrial infrastructure needed to support them, making over-subscription a persistent problem.

This over-subscription means that the round-trip times experienced by packets will be affected not only by the propagation delay, but also by the scheduling and queuing delays imposed by the Earth-based resources. Thus packets to a given destination may have to be queued until the next scheduled contact period, which may be hours, days, or even weeks away. While queuing and scheduling delays are generally known well in advance except when missions need emergency service (such as during landings and maneuvers), the long and highly variable delays make the design of timers, and retransmission timers in particular, quite difficult. This again forms a point of departure from the current Internet model, as IPN-aware applications will probably need ways to track the status of a communication and to apprise users of the expected delay before a response can be expected. This will be complicated once the IPN moves from its initial Earth-centric approach to a peer-to-peer network, since notifying users of the progress of their communications will itself consume precious bandwidth within the network.

The combined effects of large distances, the expense and difficulty of deploying large antennas to distant planets, and the difficulty in generating power in space all mean that the available bandwidth for communications in the IPN will likely be modest compared to terrestrial systems. Data rates on the order of hundreds of kilobits per second to a few megabits per second will probably be the norm for the next few decades. Another characteristic prevalent in today's deep-space missions is bandwidth asymmetry, where data is transmitted at different rates in different directions. Current missions are usually designed with a much higher data return rate (from space to Earth) than command rate. The reason for the asymmetry is simple: nobody ever wanted a high-rate command channel, and, all else being equal, it was deemed better to have a more reliable command channel than a faster one. This design choice has led to data rate asymmetries in excess of 100:1, sometimes approaching 1000:1. A strong desire for a very robust command channel will probably remain, so that any transport protocol designed for use in the IPN will need to function with a relatively low bandwidth outbound channel to spacecraft / landers.

The difficulties of generating power on and around other planets will also result in relatively high bit error rates. Current deep-space missions operate with very high bit error rates (on the order of 10e-1, or one error in ten bits) that are then improved using heavy coding. The tradeoffs between coding, bit error rate, and reliability requirements will need to be reexamined in the context of the IPN.

Finally, interplanetary communications will, at least in the near future, be characterized by intermittent connectivity between nodes. As satellites or moons pass behind planets, and as planets pass behind the sun as seen from Earth, we lose the ability to communicate with them. This effect adds to the delays experienced by packets,

and could push queuing delays to several weeks or a month if the source and destination are in opposition (on opposite sides of the sun). Inter-layer signaling, especially from the link layer to provide notifications of such breaks in connectivity, will probably be required.

We see the IPN growing outwards from Earth as we explore more and more planets, moons, asteroids, and possibly other stars. Thus there will always be a fringe to the fabric of the IPN, an area without a rich communications infrastructure. As a result, the data rate, connectivity, and error characteristics mentioned above will probably always be an issue somewhere in the IPN. For the more highly developed core areas of the IPN, it is interesting to note that delay is the only truly immutable characteristic that differentiates the IPN from terrestrial communications. Data rates, intermittent connectivity, and bit error rate can all be mitigated or eliminated by adding additional infrastructure, in theory if not in practice. Additional infrastructure can also mitigate the scheduling and queuing delays mentioned above, but the propagation delays will remain unless and until we find a way to transmit information faster than the speed of light.

These environmental characteristics: long delays, low and asymmetric bandwidth, intermittent connectivity, and relatively high error rate make using unmodified TCP/IP for end to end communications in the IPN infeasible. Using the equations from Mathis, et al [ref: http://www.psc.edu/networking/papers/model_ccr97.ps], we can calculate an upper bound on the sustainable throughput of a TCP connection, taking into account TCP's congestion avoidance mechanisms. Even if only 1 in 100 million packets are lost, a TCP connection to Mars is limited to just under 250kbps. If we assume that 1 in 5000 packets is lost (this figure was reported by Paxson as the packet corruption rate in the Internet ref: ftp://ftp.ee.lbl.gov/papers/vp-thesis/dis.ps.gz caution: very large file) then that number falls to around 1,600bps. These values are upper bounds on steady-state throughput; since the number of packets in a connection will generally be under 10,000, TCP performance would be dominated by its behavior during slow-start. Even when Mars is at its closest approach to Earth, this means that it would take a TCP nearly 100 minutes to ramp up to a transmission rate of 20kbps. Lab experiments using a channel emulator and standard applications show that even if TCP could be pushed to work efficiently at such distances, many applications either rely on several rounds of handshaking or have built-in timers that render them non-functional when the round-trip-time is pushed over a couple of minutes. It typically takes eight round trips for FTP to get to a state where data can begin flowing, for example, and an FTP server may time out

and reset the connection after 5 minutes of inactivity. This means that a conformant standard FTP server could be unusable for communicating even with the closest planets.

1.3. A "Postal" Communications Model

We have concluded that the standard Internet protocols should be essentially "terminated" at the Interplanetary Gateways and the information payloads conveyed through a new set of protocols better suited to the long distances, variable delays and asymmetric data rates of the interplanetary backbone network. In essence, the design is analogous to a kind of postal relay system in which messages are delivered to the intermediate Interplanetary Gateways, extracted from their standard Internet protocols, and encapsulated in new link and transport protocols to be forwarded to the next IPN gateway and ultimately into the target internet.

Internet electronic mail already works in this fashion but the transfer of files, the operation of the World Wide Web, and remote interactive applications do not fit into this model directly. There are circumstances under which researchers on planet Earth do need an ability to interact with remote devices, for example to steer wheeled robotic vehicles around on the surface. But because of the enormous round-trip delays, such systems must work very indirectly. For example, to steer the Mars Pathfinder rover, one sends instructions about intermediate points that the robot must steer past. This is analogous to automatic airplane pilots that are given a series of coordinates through which to pass. In effect, a planned itinerary is sent and the robot vehicle executes the plan, dealing with local conditions as required.

The "store-and-forward" nature of this communication method is reminiscent of bucket brigades, except that the contents of the buckets are actually the payloads (i.e. data) of the applications that utilize the network. The concept of "custody" is important in such a system. A sender does not relinquish a copy of a transmission until it is sure that the next in line has successfully received it.

2. IPN Architectural Overview

We now consider five broad areas that represent areas of significant research in the Interplanetary Internet architecture:

* The communication conducted between independent internets (termed "Inter-internet dialogs")
* The architecture and functions of the unique nodes of the Interplanetary Internet
* A security architecture for meeting anticipated data and infrastructure protection needs
* The issues in developing a stable backbone network for the Interplanetary Internet

* The issues in deploying internets on remote planets, asteroids, and spacecraft

The following five sections of this document consider each of these areas. Following these sections we present our conclusions to-date.

3. Inter-Internet Dialogs

This section first presents four principles that guide the design of the Interplanetary Internet. In doing so, we introduce the concept of the "bundle" layer, a protocol layer providing end-to-end service in the IPN. The section continues by discussing the information carried by the bundle layer, and concludes by discussing reliability at the bundle layer.

3.1. Principles of Design

3.1.1. Name Tuples Consisting of Administrative and Routing Parts are the Means of Reference

In the (terrestrial) Internet, names are administrative in nature, and are hierarchically organized. The Domain Name System (DNS) uses a highly distributed database to translate the name to a numeric address, and addresses are the common medium used throughout the network for reference. This scheme has worked well in the Internet, but the emergence of network address translators and other partitioning mechanisms have begun to cause some problems with this scheme.

One of the problems involved with using DNS names across interplanetary space is the distributed nature of the DNS database. This means that it is entirely possible for the portion of the database that can resolve a name to its address to be far (very far, in the Interplanetary Internet) from the host requesting resolution. This means that the times required to resolve names to addresses can become impossibly high, especially when the issues of intermittent connectivity come into play. The DNS has a mechanism to replicate portions of its database, using a technique known as zone transfers. However, this is not a good solution in the Interplanetary Internet, either. One could easily spend all of the available communication time transferring the ".com" zone to another planet, rather than actually transferring data. Clearly, another approach is indicated.

In our initial designs, we considered creating a new top-level domain, e.g. ".sol", and assigning to it topological significance. We constructed a scheme by which we routed on the "most significant" portions of the domain name, such as ".mars.sol", and essentially bet that these portions would be topologically significant, rather than administratively significant. This scheme, while attractive from the standpoint of making use of existing infrastructure, makes use of that infrastructure in a bad way. We were forced to "grandfather" existing top-level domain names to be bound to Earth's Internet, so

that ".com" meant ".com ON EARTH." This spawned philosophical debates within the group regarding sovereignty and the current top-level domain structure within the internet, but also had the potential of creating serious technical problems. For example, some

organizations encode geographic information at fairly low levels of their DNS names (consider "zurich.ibm.com", and then consider "mars.ibm.com"). If all ".com's" are shipped off to Earth, the "mars.ibm.com" data is going to take a serious detour. We eventually concluded that this was not the right approach.

We have concluded that names in the Interplanetary Internet should consist of a tuple that contains the administrative part plus a routing part. These names must be carried end-to-end throughout the Interplanetary Internet. The routing part serves the purpose of the new top-level domain described above, except that it is not required to conform to the naming conventions of the Domain Name System (i.e., it may be numeric rather than textual), it may be hierarchically organized, and it must be routed. This requires us to develop the means for computing and distributing routing information, but relieves us from a dependence on the relationship between administrative names and network topology.

3.1.2. The Routing Part of an IPN Tuple Identifies an Internet

The routing portion of the name identifies an IPN Region. We envision the IPN as a "network of Internets", and the IPN Region, to some extent, allows us to route to a particular Internet. The use of an IPN Region is an explicit form of aggregation that is not otherwise possible using administrative names.

It is necessary and sufficient that the administrative portion of an IPN name be resolvable to a useful address within its IPN Region. We are currently treating the structure of the routing part of the name in a manner similar to the structure of DNS names: the name space of the routing part is a tree of text labels separated by "dots," with the root node of the space having a null label. We denote a name in the IPN in the following manner: {administrative part, routing part}. So, if "earth.sol" were an IPN Region encompassing the entire Earth, the web site for the Internet Society's IPN Special Interest Group would be { www.ipnsig.org, earth.sol}.

In order that any node in the IPN be able to send data to any other node, the routing part of the name must be interpretable everywhere. That is, any IPN node, when confronted with any valid IPN Region name, must be able to identify a transport layer destination that moves the data toward that region.

3.1.3. The "Bundle Layer" Terminates Local Transport Protocols and Operates End-to-End

In the IPN, we cannot ever assume that there is direct connectivity between source and destination. That is, we cannot assume that bits

emitted by a source can travel, delayed only by routing and transmission delays, to the destination. There may be any number of reasons for this, ranging from physical (the destination is on the far side of a distant planet and can't communicate with anything

right now), to schedule-related (a required IPN gateway is currently serving other customers), to administrative (the source is only on during the day, the destination only during the night). For the long-haul links of the backbone, information will almost certainly have to be stored for some amount of time as the antennas used for the long-haul links will almost surely be highly directional.

Thus depending on the schedules of the nodes involved and the possibility of high-priority interrupt traffic, the nodes that make up the IPN may have to buffer data for hours, days, or weeks before it can be forwarded. Also, the highly varying communications environments that will make up the IPN, ranging from optical fiber on Earth to wireless communications around Mars, to the long-haul links of the backbone, suggest that different transport protocols will be needed for the different environments. It makes sense, therefore, for the IPN nodes to terminate the transport-layer protocols used in the respective IPN regions, holding data at a higher layer before forwarding it on, possibly using a different transport-layer protocol.

We call this higher layer the "bundle layer," and the protocol used to send data between the various nodes of the IPN the "bundle protocol." The term "bundle" is used to connote the store-and-forward aspect of communications where as much interactivity as possible has been distilled out of the communication. A bundle file transfer request, for example, might contain the user's authentication (login/password, e.g.), the location of the file to get, and where that file should be delivered in the requester's IPN domain. All of this information would be transmitted as one atomic "bundle," and the requested file would be returned. We use the term "bundle" rather than "transaction" to avoid notions of two-phase and three-phase commitment that are commonly associated with transaction processing.

In traditional networking terminology it is generally the transport-layer protocol that operates end-to-end. Since IPN nodes terminate transport layer protocols in order to buffer data and to enable them to use a transport protocol appropriate to the IPN region through which data will be sent, it is the bundle layer in the IPN that operates end-to-end.

It should be noted that terminating the transport protocols at the IPN nodes decouples the internets in different IPN regions to a significant degree. This has the desirable effect of also decoupling the evolutionary rates of those internets: changes in the Earth's Internet do not necessarily dictate changes in other internets. This is important in an environment in which resources are and will

continue to be severely constrained.

Figure 1 illustrates the progression of a bundle of data through the Interplanetary Internet, from its source at host "A" to the destination at host "E." Custody transfers are indicated by

asterisks (*), and occur at B, C, D, and E. Host "A" initiates the bundle transfer, and the bundle is transferred using internet protocols (i.e., TCP and IP) to bundle node "B", which serves as the gateway between the left-hand internet and the interplanetary backbone. The box icon indicates that custody of the bundle is transferred to that gateway. When conditions permit, the bundle is forwarded on to the next hop in the store-and-forward chain. In this case the next hop is another host within the interplanetary backbone network: host "C".

Also illustrated in Figure 1 is the notion of a "return receipt" sent by the ultimate destination of the data back to the source. This is an optional service, much like a return receipt within the postal system. If the source desires notification of delivery, that is accomplished by a separate return receipt, which is transmitted as its own bundle, and is subject to the same custody transfers as the original transmission (similar to the fact that a postal return receipt is, in itself, a postcard). It is shown figuratively as bypassing the forwarding path (E to D to C to B to A), but this is simply for clarity. The return receipt is forwarded in exactly the same manner as the original data is.

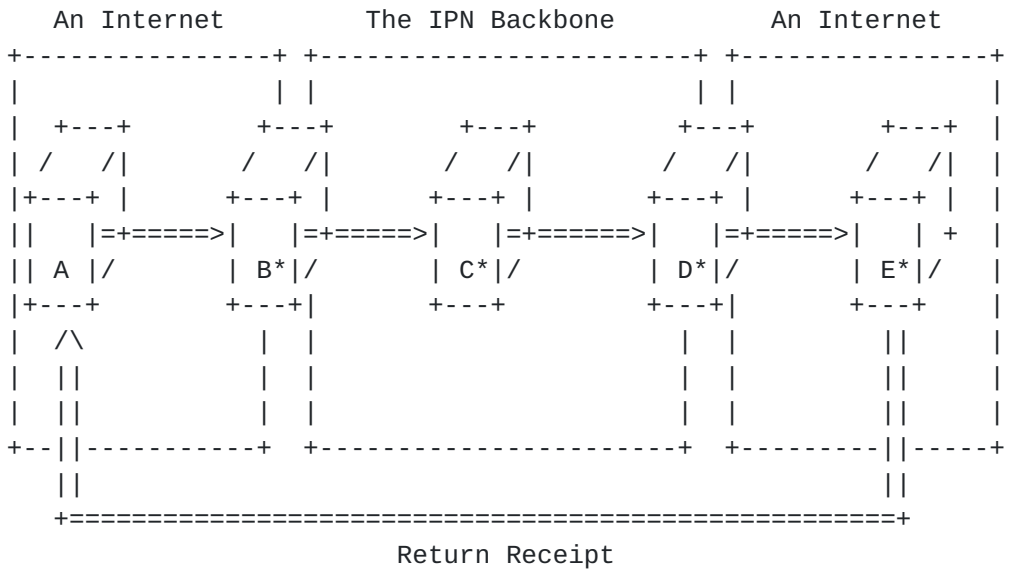


Figure 1. Custody Transfers

Many aspects of this mode of data transfer resemble the postal system, or even the Pony Express. The source depends on the store-and-forward nodes in the chain to operate on its behalf to deliver data that may not be retained at the source. Source notification

that the destination has received the data is optional, and there is no guarantee or even any firm expectation on the part of the source about when the data will be delivered.

Conceptually, the bundle protocol resides above the transport

layer(s), and operates end-to-end between the ultimate source and the ultimate destination of the data. The bundle layer provides a number of services to applications using it. For example the bundle layer carries the source and destination name tuples end-to-end to support the late binding of the destination name's administrative part to an address. The bundle layer also carries the users' specifications for reliability, quality of service, and security. Because the communication resources are precious, it is desirable to provide error recovery mechanisms that do not necessitate discarding of bundles that contain errors. We are considering adding to the information carried in a bundle some information to assist in transfer-related error handling. We are thinking in terms of active networking, using a combination of active packets and active nodes as a means of specifying appropriate actions to take in the event of problems in completing the transfer. Additionally, the bundle layer provides invoking applications with a transfer identifier that is carried with the bundle, and is used in "return receipts." [Ref: http://www.comsoc.org/pubs/surveys/1q99issue/psounis.html]

3.2. Information Carried by the Bundle Layer

To effect the end-to-end transfers necessary in the IPN, the bundle layer must carry some information end-to-end. This section documents our current thinking on the information that must be carried end-to-end, and notes which of those data elements may be supplied by the application using the bundle service.

* Bundle Identifier: this is a monotonically increasing number that is carried in the bundle, and also returned to the application to support return receipt processing. It is not necessarily a sequence number, and as a result, there is no requirement that the value of Bundle Identifiers increase consecutively. The requirement for monotonicity derives from a need to provide robustness against system crashes, and therefore must be persistent across system crashes.
* Remote entity name: this is the IPN name of the remote bundle agent, and is a tuple, as described above. It is supplied by the local application using the bundle service.
* Source entity name: this is the IPN name of the local bundle agent, and is a tuple. It is supplied by the local bundle service, since a particular host may have multiple names and one may be chosen based on routing decisions or other criteria opaque to the application (as in multihomed hosts). The source name may be returned to the application to support return receipt processing.
* Authentication information: this is information, such as a digital signature, that is passed by the application to the

bundle layer to unambiguously identify the source of the bundle. (Just what the source of the bundle is, person, place, address, etc. is still undecided.) This information is checked for validity at both the source bundle agent and the destination bundle agent. The authentication information may also be used

for access control purposes within the network.

* Source application instance handle: this is similar in nature to a source port number in that it identifies the sending application. Since bundles are inherently non-interactive, the typical use for this handle is to "reanimate" the source application when a return receipt arrives. This could be hours, days, or weeks after the initial transmission, so the handle may well be a reference to a structure that allows the application to be reinstantiated with known state. The source application instance handle may be used at the destination as an identifier, but may also be redundant with the end-to-end authentication information for this purpose. This handle is supplied by the source application.
* Destination application instance handle: this is essentially the destination port identifier. As with ports, these must be known to and supplied by the source application. We have not yet fully explored the implications of port advertisement across the IPN.
* Size of data: this is a statement of the size of the bundle, in bytes. It is supplied by the source application, and is used initially to ensure that sufficient space is available to store the bundle for its initial transmission. Nodes receiving transmitted bundles use this information in the same manner: as a means of making a determination about the availability of storage early in the bundle handling process. In forming the initial bundle, the bundle layer at the source may use the size of data parameter as a consistency check on the amount of data actually delivered.
* Handling instructions: these are parameters supplied by the user with the bundle that convey the user's preferences to the network. Our thoughts on just exactly what these parameters look like are not yet firm, however, our thoughts are that they include some or all of the following: priority, quality of service (although we are still debating what this means in the context of the IPN), elapsed time after which the content of this bundle is meaningless (time to live as specified by the user), reliability requirements, and any error handling information. For the most part, these are requests, and we perceive that the bundle layer may either override some or all of these requests or fail the request if local policy does not permit that particular user to make that particular request at that particular time.
* Data Descriptor: This is a reservation token that is generated by the bundle layer. Its detailed definition and use are still to be determined.
* Time to Live: This is the time after which this bundle is to be discarded from the network. It is present to facilitate the

recovery of network resources and to terminate routing loops, should they occur.

* Loose/Strict Source Route and Record: This information is provided by the source application to facilitate debugging of the network. It consists of a list of names of IPN nodes

through which the bundle must pass on its way to the destination, and our intent is that it should behave similarly to the corresponding option within IP.

* Current bundle custodian: the bundle protocol supports store-and-forward operation in which the custody of a bundle (that is, the responsibility for ensuring reliable delivery) may transfer from one IPN node to another as the bundle progresses through the IPN. There is not a requirement for each IPN node encountered to assume custody of a bundle. As a result, it is necessary to identify the upstream node that has custody of the bundle, in order to either request retransmissions or to accept custody of the bundle.
* User data: this is intended to be all of the data that the remote entity requires to perform whatever operation is requested. Since the environmental characteristics of the IPN make interactivity difficult, the notion is that all of the information that is required to perform a particular "transaction" would be provided in a single bundle.

3.3. Reliability at the Bundle Layer

Because no single transport-layer protocol operates end-to-end across the IPN, end-to-end reliability can only be assured at the bundle layer. At each node along an end-to-end route, the bundle-layer protocol entity passes bundle data to the Transport layer for transmission. Each bundle layer entity is highly confident that the transport layer will successfully convey the data entrusted to it to the next bundle-layer protocol entity (which may "take custody" of the data or merely relay it; a single hop). But failures are possible (e.g., a host computer does an unplanned reboot). Just in case the highly unlikely happens and a Transport-layer transmission fails, the first subsequent node that detects the failure and is capable of taking custody of the bundle will request that the prior custodian re-transmit any missing data (again using Transport-level transmission services across, potentially, one or more relay nodes).

The bundle layer's confidence in the effectiveness of the underlying Transport-layer protocols is reflected in the design of the timers for bundle-layer reliability. These timers are highly optimistic _ that is, they expire as late as possible _ in order to give the Transport protocols every opportunity to complete reliable transmission. The effect of this optimism is to minimize the chance of unnecessary bundle-layer retransmission, which could seriously degrade IPN performance by consuming valuable bandwidth.

3.4. Bandwidth Allocation via Market Mechanisms: "Starbucks"

To promote effective and efficient use of the IPN's scarce transmission resources, some sort of sophisticated and adaptable bandwidth allocation system will probably be needed. The scheme described below is based on a free-market notion of "fare-paying packets", where at initial transmission each bundle is issued a

"draft" on some small percentage of IPN resources. These resources might well map back to actual monetary funds provided by the originator of the bundle to the provider(s) of the IPN service. The bundle in effect pays its own way across the various legs of end-to-end transmission. As it traverses each hop, the bundle spends funds from its original draft until either it is received or else its funds are exhausted. If a bundle is dropped due to insufficient funds then the hope is that all available transmission resources were allocated to bundles that were allotted more funds and therefore were presumably more important (to somebody).

At the initial Send-bundle service request, the source application (bundle sender) would specify total funds allocated to getting the bundle delivered to the destination. Total funds allocation needs to be a function of (a) the total number of bytes of data and metadata to be sent, and (b) the prices charged for each "transmission class," (something like First Class, Business Class, Economy Class, Steerage, Overhead Bin; etc.). The allocation should cover the anticipated costs of traversing all the bundle hops along the anticipated route to destination. If the bundle must be split up into multiple packets (bindles, segments), the bundle agent that performs the split also distributes the bundle's total funds among individual packets. This distribution will probably be on a prorated basis.

Also, the source application would supply the bundle (and, by implication, each of its packets) with traveling instructions:

For each time epoch that elapses while awaiting transmission from any single bundle transmission agent:

* The length of the epoch in units of time (seconds?)
* The queue (transmission class) to wait in over the course of the epoch

For example: get in the Steerage queue, but if 30 minutes pass and you still haven't been transmitted then pay for an upgrade to Business Class; if you still haven't been transmitted after 20 minutes in Business Class, then give up.

At each bundle transmission agent, each packet is handled as follows:

* For each of the packet's authorized time epochs until the packet is transmitted (that is, initially handed to the underlying communication layer; retention until successful custody acquisition by downstream agent is provided at no charge):

  - If (epoch duration * packet length * agent's price per unit of time per byte for this epoch's transmission class, i.e. queue)

is greater than packet's residual funds, then discard the packet. The packet's residual funds remain unexpended.

- Else, append the packet to the requested queue. Then:

- If epoch expires before packet is de-queued for transmission, then charge the source application an amount equal to (epoch duration * packet length * price per unit of time per byte for queue), reduce packet's residual funds by this amount, and start the next epoch; if no more authorized epochs, give up.

- Else, upon de-queuing the packet for transmission, charge the source application an amount equal to (length of time spent in queue * packet length * price per unit of time per byte for queue) and reduce packet's residual funds by this amount.

* Transmission classes _ queues _are of varying maximum length (the more expensive queues are shorter) but packets in all transmission classes are at the same level of priority; packets are de-queued in round-robin fashion to ensure that no class is altogether starved for service. Because First Class is a shorter queue than Business Class, packets that pay for First Class spend less time waiting.

* Separately, an Emergency queue is provided for system-critical packets. Everything in the Emergency queue has higher priority than everything else, so nothing else gets transmitted until the Emergency queue is emptied.

Periodically, each bundle protocol agent reports aggregate charge amounts back to the source applications and also to some central accounting authority ["the bank", nominally based on Earth]; this is a separate application-layer protocol. When the central authority determines that a source application is out of funds, it reports the source application's bankruptcy to all bundle agents; from that time on, all service requests and packets received from that source application are rejected.

Although this particular scheme may ultimately prove too complex to be workable, we think the general principle of fine-grained bandwidth allocation could contribute significantly to the viability of the IPN.

4. IPN Nodes

Nodes within the IPN have a number of responsibilities. As members in a store-and-forward chain, they have the responsibility for resource allocation to support bundle transfers. These resources include, among other things, buffer space and transmission capacity.

Additionally, IPN nodes have the responsibility of actually executing the bundle transfer. Reliability requirements for bundle transfers are specified by the using application, and include both reliable and unreliable transfers (possibly with some intermediate, or partial,

reliability services). The IPN nodes are responsible for using whatever reliability mechanisms exist in the underlying (transport-and-below) layers, and augmenting those mechanisms as necessary to effect the required reliability.

Finally, IPN nodes are responsible for routing bundles between IPN domains. IPN nodes may depend upon the services of the local internets (or the IPN backbone) for intra-domain routing.

In this section, we first briefly state the types of IPN nodes that we have identified, and then we provide a number of exemplary end-to-end data transfer descriptions. Finally, we list the error conditions that we have identified that may occur at the bundle layer during the course of end-to-end data transfer.

4.1. Types of IPN Nodes

We identify three grades of IPN functional capability. In order of increasing scope, they are: agent capability, relay capability, and gateway capability. All IPN nodes are able to act as bundle agents; some bundle agents are additionally able to act as IPN relays; some IPN relays are additionally able to act as IPN gateways.

* Bundle agents build and consume bundles. These could be stand-alone proxy devices or could be an operating system service collocated with an application. The endpoints of an end-to-end bundle transmission need not be any more than bundle agents, though they may additionally have relay or even gateway capability.

* IPN Relays receive bundles and forward them toward their destinations, either within or between IPN regions.

* IPN Gateways reside at the interface between IPN regions. The IPN gateways perform routing between the IPN regions.

Orthogonal to these grades of capability is the ability to take custody of a bundle. The endpoints of a bundle transaction are typically the initial and final custodians of the bundle. Non-gateway relays may take custody of the bundles they receive; alternatively, they may simply provide mitigation of R2 effects (signal strength losses due to the extreme distances of interplanetary communication), but provide no custody transfer capabilities. (In the latter case they are essentially "repeaters.") Gateways normally take custody but are not required to do so.

4.2. Example end-to-end transfer

We provide the following example of an end-to-end transfer that crosses multiple IPN Regions: A host on Earth sends a bundle to a destination on Mars. Figure 2 illustrates the network that is the subject of this example _ the Interplanetary Internet in this example

has five distinct regions interconnected by four IPN Gateways. This example highlights the actions taken by the bundle layer and the interactions of the bundle layer with applications and with underlying transport protocols.

4.2.1. Backbone connectivity

It is important to have some understanding of the routing that is required at the IPN Gateways. Unlike terrestrial communications, the IPN's long-haul communication links are directional, mobile, and highly scheduled. This is important, because directionality combined with mobility means that a transmitter and receiver must track each

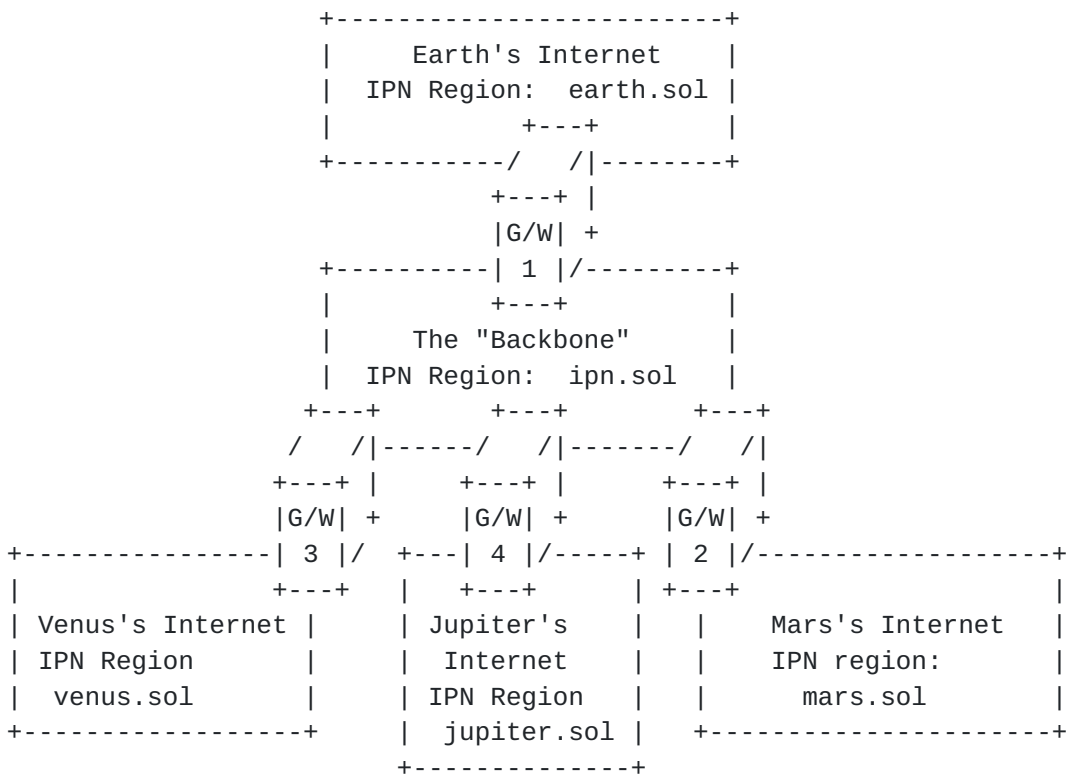


Figure 2. An Interplanetary Internet of Five IPN Regions

other in order to establish and maintain a communication link. In the IPN, much of the mobility is due to orbital mechanics and is therefore relatively predictable. However, this means that nodes that we would normally consider to be fixed, such as antennas on the surface of the Earth, are actually highly mobile as a result of the Earth's rotation and its revolution around the Sun. (In this example, we confine ourselves to our local solar system, and do not consider the motion of our sun relative to celestial bodies outside our solar system.) We can describe the predictable aspects of node

mobility with an ephemeris, a table of the positions of celestial bodies at specified intervals of time. Both a directional sender and receiver must know the other's position in order to establish a link between the pair. In addition, these communication resources are highly scheduled. It is not sufficient for a receiver to point at a

prospective target and just wait _ for example, a terrestrial node will typically have to point at several targets sequentially, and an interplanetary node will typically not have enough power to just wait for incoming messages. Rather, a schedule of communication opportunities must be calculated and then refined with planned communication instances. A communication opportunity establishes that the endpoints could establish a link if they were pointing at each other at the proper times. We refer to a planned communication instance as an agreement (although not irrevocable) between the two parties to establish contact and communicate for a defined period of time. The protocols for establishing the schedule of communication instances between all pairs of possible communicants will evolve -- from something primarily done manually to something more automated as the Interplanetary Internet grows.

The scheduled nature of connectivity in the Interplanetary Internet, particularly across the deep-space links, means that at the time of a bundle's arrival at an IPN Gateway, some or all of the possible outbound routes may be "down." The gateway must store the bundle until the appropriate link is available and then transmit the bundle over that link. One of the fundamental differences between the Interplanetary Internet and the terrestrial Internet is this inherent use of store-and-forward mechanisms in routing bundles. With that in mind, let us consider the routing decisions made at some of the IPN Gateways in Figure 2.

4.2.2. IPN Gateway routing

Bundle routing at an IPN gateway will typically have to deal with a mix of continuously available links and intermittently available links. Routing across a continuously available link is a relatively straightforward activity. Routing in the presence of intermittently available links can be significantly more complex, as the gateway will need to decide not only the next hop destination but also the time at which to send the bundle. Conditions elsewhere in the network may make it more desirable to send a bundle to a next-hop destination that is not yet accessible, even when an alternative route is currently available. The routing function in IPN Gateways and other IPN nodes that have intermittent links has three distinct parts: the contact scheduler, the route evaluation algorithm, and the dispatcher algorithm.

Figure 3 illustrates the relationship of these functions with each other and with other elements of the communication protocol suite. The contact scheduling application establishes the schedule for communicating with next-hop neighbors. The implications of this scheduling activity are broad _ orbital mechanics, resource

management onboard the spacecraft, prospective communications loads, and so forth all play a role. Because spacecraft resources must be coordinated among all functions (not just communications), this is typically going to be part of a larger resource management application (interactions with functions such as pointing and power

control are not shown in Figure 3, but are present nonetheless). It is very likely that the contact scheduler will *not* be local to the bundle node, but rather a centralized function that distributes a contact schedule to IPN nodes that perform routing. This may change in the future to a distributed contact scheduling algorithm, but for the foreseeable future, this will not be the case. The product of the contact scheduler is a schedule of planned contacts, durations, expected data rates, etc. It is intrinsically link-state in nature.

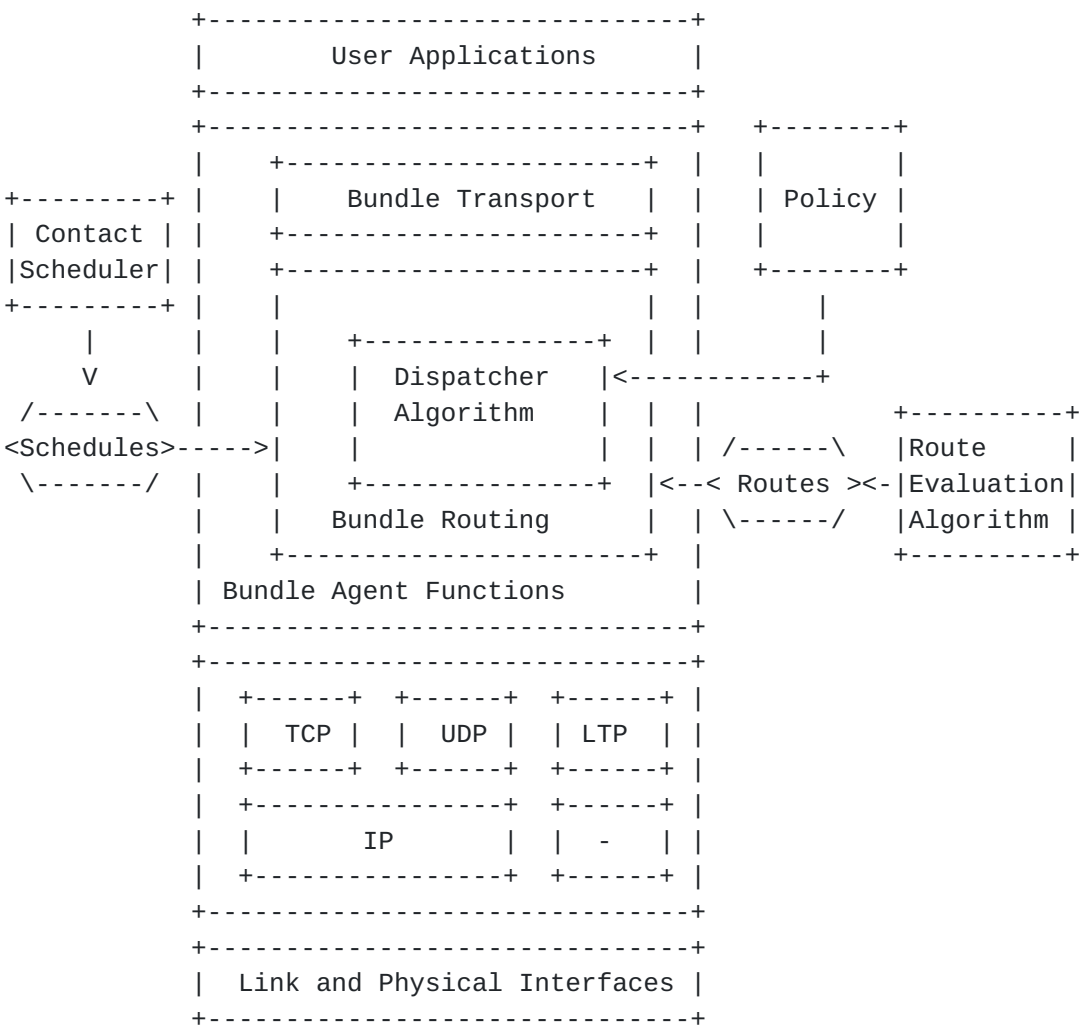


Figure 3. IPN routing applications and their relation to other communication functions.

The route evaluation application exchanges information with all first-hop neighbors (we hope this occurs infrequently) to build a general picture of the IPN beyond the first-hop neighbors. We

currently view this in terms of a distance-vector representation, but with metrics such as the expected delay to the destination IPN region and average aggregate data rate to the destination IPN region. The mechanics for building these metrics are still in development.

The dispatcher algorithms accepts routing requests from the bundle transport layer and builds a "manifest" for each next-hop contact that is subsequently consumed by the bundle routing layer. The dispatcher application consumes some or all of the following in deciding how to schedule bundle transmissions: the contact schedule, the routing information, local policy information, and the per-bundle specifics provided by the bundle transport layer (such as bundle destination, length, priority, time-to-live, and possibly "Starbucks"-related information). We envision a family of different dispatcher algorithms, operating as "plug-ins," that provide different levels of sophistication in the scheduling function to suit different needs. The simplest versions of the dispatcher might just consider destination to schedule the bundle on the soonest-possible outbound contact. More sophisticated versions might consider priority, bundle length (to preserve atomicity), and time-to-live requirements. Others could support the Starbucks model or apply optimization techniques to attempt to improve the use of each contact.

The following is a conceptual description of what happens: when a bundle arrives at the bundle layer and needs to be routed, the bundle routing function posts a request to the dispatcher, noting the destination, length, priority, time to live, etc. of the bundle to be routed. The dispatcher integrates this bundle into its manifest and, at the bundle's transmission time, informs the bundle routing function to send the bundle to the appropriate next-hop destination.

4.2.3. Systems participating in example bundle data transfer

Figure 4 is a revision of Figure 2 to highlight those portions of the Interplanetary Internet that participate directly in the bundle transfer example. Also shown in Figure 4 are the source and destination for the bundle data transfer, and the Domain Name System equivalents in the terrestrial Internet (DNS 1), in the IPN "Backbone" (DNS 2), and in the Mars Internet (DNS 3). This figure will serve as the basis for the following bundle data transfer example.

Table 1 provides the full host names of each of the primary elements in Figure 4.

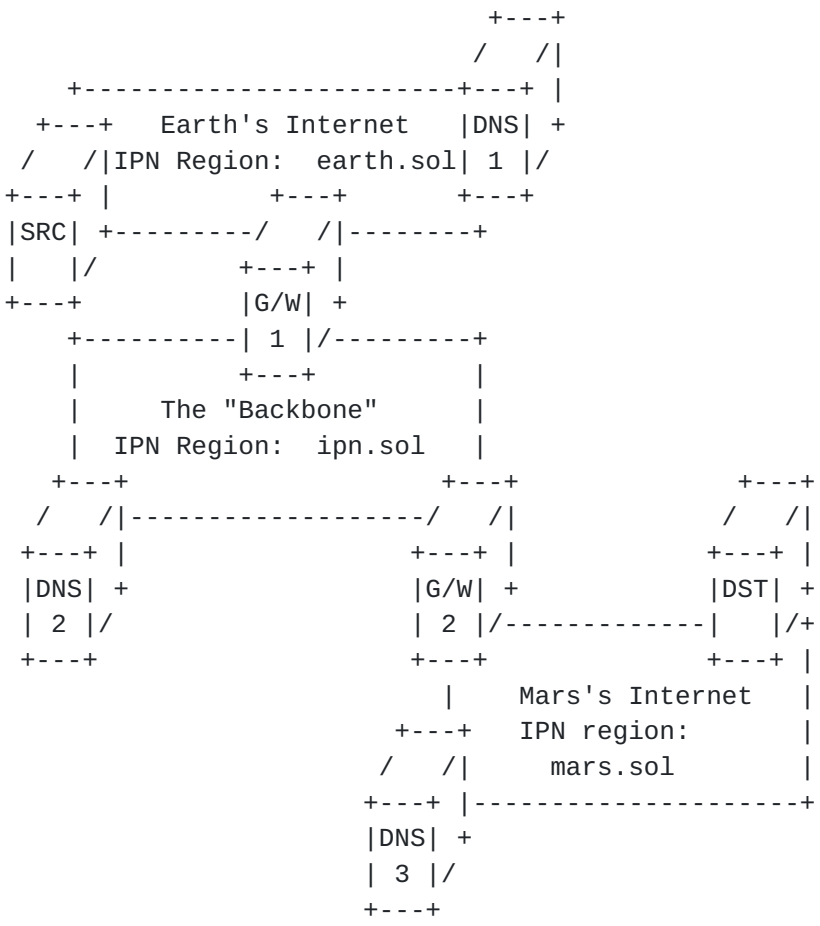
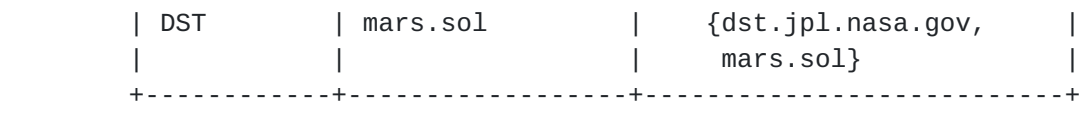


Figure 4. Interplanetary Internet showing principal systems.

Table 1. Host name tuples.

Host	IPN Regions	Host name tuples
SRC	earth.sol	{src.jpl.nasa.gov, earth.sol}
IPN G/W1	earth.sol	{ipngw1.jpl.nasa.gov, earth.sol}
	ipn.sol	{ipngw1.jpl.nasa.gov, ipn.sol}
IPN G/W2	ipn.sol	{ipngw2.nasa.mars.org, ipn.sol}
	mars.sol	{ipngw2.nasa.mars.org, mars.sol}


| DST | mars.sol | {dst.jpl.nasa.gov, mars.sol}

4.2.4. **Step 1:** Bundle creation and first-hop transmission

An application on the source host in Figure 4 has data that it wishes to send to the destination on Mars. The exact content of this data is opaque to the bundle transfer, but assume that it contains all of the information necessary to accomplish some desired function. That is, assume that application-specific instructions for storage, handling, error processing, and disposal accompany whatever data object is to be operated upon. The application invokes the bundle agent, supplying it the information shown in Table 2.

Table 2. Information passed from source application to bundle agent.

Item	Value	Description
Destination host name	{dst.jpl.nasa.gov, mars.sol}	IPN Name tuple of the destination
Destination application instance handle	0x00000008	Similar to port number. Can be "well-known" (i.e., identify a daemon) or "ephemeral" (i.e., identify a running, suspended, or hibernating process
Source application instance handle	0x1763421A	Value used to identify the appropriate instance of the source application for response processing
Handling instructions	Reliable delivery, normal priority, data obsolete in 36 hours.	The services requested from the bundle layer.
User data	N/A	

The bundle agent creates a bundle and stores it in persistent storage (on disk or other non-volatile memory). There are some fields of the bundle header that the bundle agent must supply: the Bundle Identifier, the Source Host name tuple, the Custodian name tuple, and the time to live. (The application may state a time after which the data are obsolete, but the actual time-to-live field in the bundle header uses the application's data in combination with network

restrictions on time-to-live to initialize this field properly.) The bundle agent's routing function requests a route from the dispatcher application, and receives next-hop destination information and source information. (Since a host may reside in multiple IPN Regions, the source host name tuple is a function of the outbound route selected.

The bundle agent uses this information to complete the Source Host and Custodian name tuples prior to transmission.)

Note: For hosts that have continuously available communication resources, it is likely that an optimization would be implemented to reduce the required interaction with the dispatcher application. In this instance, the bundle routing function might consult a local routing table before contacting the dispatcher, and only requesting routing information from the dispatcher if there is no appropriate entry in the routing table.

The bundle in this example is destined for the "mars.sol" IPN Region (per Table 1). The dispatcher (or routing table) determines that the proper next-hop destination for the mars.sol region is {ipngw1.jpl.nasa.gov, earth.sol}, and that the appropriate transport protocol to use is TCP. The bundle agent consults the Terrestrial Domain Name Server to resolve ipngw1.jpl.nasa.gov to an IP address, establishes a TCP connection with ipngw1, and transfers the bundle to it. The bundle agent at the source retains a custodial copy of the bundle in persistent storage.

4.2.5. **Step 2:** Bundle processing at first-hop destination

When the IPN Gateway {ipngw1.jpl.nasa.gov, earth.sol} receives the bundle via TCP, it stores it on persistent storage (disk). The bundle agent consults the dispatcher, and is informed that the appropriate next-hop destination is in the "ipn.sol" IPN Region: {ipngw2.nasa.mars.org, ipn.sol}. The dispatcher provides the time at which the bundle should be transmitted, at which time the contact scheduler and its associated functionality will ensure that there is a contact with ipngw2 via the Long Haul Transport Protocol (LTP). In considering alternative contacts for the bundle, the dispatcher checks the time-to-live in the bundle, which was 36 hours from the time of initial submission to the bundle agent at the source, to ensure that the route selected is consistent with the time-to-live requirements of the bundle. The bundle transport functionality of the bundle agent in ipngw1 accepts custody of the bundle, updates this information in the bundle header, and informs the source that has done so. The sources bundle agent deletes its custodial copy of the bundle. When the time indicated by the dispatching function arrives, the bundle is transmitted via LTP to the IPN Gateway that connects the ipn.sol Region with the mars.sol Region: {ipngw2.nasa.mars.org, ipn.sol}.

4.2.6. **Step 3:** Bundle processing at gateway to destination IPN Region

The Mars gateway, {ipngw2.nasa.mars.org, mars.sol}, receives the

bundle from the Earth gateway via LTP. It stores the bundle in persistent storage and accepts custody of the bundle, signaling back to the Earth gateway that it has done so. When the notification of acceptance reaches the Earth gateway, ipngw1 deletes its custodial

copy. The Mars gateway consults its dispatcher application to find an outbound contact to forward the bundle over. The dispatcher returns an indication that the appropriate next hop is the destination itself, that the proper transport protocol is TCP, and that the destination is accessible immediately. The gateway verifies that the time-to-live has not expired, and forwards the bundle to the destination.

4.2.7. **Step 4:** Bundle processing at destination

The destination bundle agent receives the bundle via TCP, stores it on its own persistent storage, and accepts custody of the bundle from IPN G/W2. The bundle agent "awakens" the destination application process identified by the Destination Application Instance Handle. This may involve creating a new instance of a server from a daemon process, signaling an idle running process, or reinstantiating a process that has been suspended with its state stored on persistent storage. (The specifics of this are system-dependent, and may have to be robust against system restarts, upgrades, and migration of processes from one host to another.) The bundle agent deletes the copy of the bundle from persistent storage when the application has received it. The destination application may generate an application-layer acknowledgment in a new bundle and send it to the source's application instance identifier (0x1763421A from Table 2).

4.3. Error Conditions at the Bundle Layer

This section describes the error conditions that may arise at the bundle layer during bundle creation and transport. When these errors occur within the sender's IPN domain, it may be possible to conduct a near-real-time dialog to correct them before the bundle is forwarded. We say `may be possible' because even if two nodes are in the same IPN domain, they may not have real-time connectivity. An example of such a situation would be if a lander were on the opposite side of the planet from its IPN gateway, and used bundles to communicate with the gateway through a low altitude orbiter, with the orbiter itself serving as a bundle agent.

Table 3: Error conditions at the bundle layer.

Error	Description	Places where Error can Occur
1*	Unknown destination region	Source Bundle Agent
2*	Invalid Source App.	Source Bundle Agent
3*	Bundle Parameter Syntax Error	Source Bundle Agent
4*	Bundle Parameter Semantic Error	Source Bundle Agent
5*	Invalid Node Name in LSRR or SSRR	Any bundle node
6	Insufficient buffer space	Any bundle node
7	DNS unreachable	Any bundle node
8*	Time exceeded	Any bundle node other than the source agent
9*	Source Entity Access Denied	Any bundle node other than the source agent
10*	Invalid Administrative Destination Name	IPN gateway serving destination IPN domain
11*	Invalid Destination App.	Destination
12*	End-to-end Access Denied	Destination

The errors that can occur at the bundle layer are shown in Table 3. Error numbers marked with an asterisk (*) are reported back to the sending application.

* Unknown Destination Region: This error occurs when the source bundle agent is directed to create a bundle destined for an IPN Region that is not recognized (i.e. one for which there is no applicable route known to the Dispatcher Application). Note that only the IPN Region part of the destination name has to be interpretable outside the destination's IPN Region. In particular, the administrative part of the destination name need not be interpretable to the source DNS (assuming the source and

destination are in different IPN Regions), so it cannot necessarily be checked when the bundle is created.

* Invalid Source Application: If the source application instance handle supplied by the source application is invalid, the source

bundle agent responds with an Invalid Source Application error. This might be the case, for instance, if the source application provided an instance handle referencing a system process for which the application didn't have privileges.

* Bundle Parameter Syntax Error: The source bundle agent may check the syntax of some of the bundle-creation parameters (i.e. it may ensure that the end-to-end and IPN access security certificates are well-formed, etc.) If a parameter is found to be syntactically incorrect or obviously and definitely erroneous, the bundle agent will report a Bundle Parameter Syntax Error back to the source that includes, at a minimum, the parameter that caused the error.

* Bundle Parameter Semantic Error: If the source bundle agent can identify a particular bundle creation parameter as being well-formed but unserviceable, it will report a Bundle Parameter Semantic Error to the source application that includes, at a minimum, the parameter that caused the error.

* Invalid Node Name in LSRR/SSRR: If an invalid node name is discovered in the loose or strict source route & record, the bundle agent that detects the error will propagate it back to the source application. Note that it may be advantageous to have bundle agents check the validity not only of the next hop in the source route but as many entries as they can. The value of checking multiple entries would be in detecting errors as soon as possible, preferably before the bundle traversed any of the long-haul links.

* Insufficient Buffer Space: If a bundle agent does not have sufficient buffer space to accept a bundle, it drops the bundle and generates an Insufficient Buffer Space error. Note that a bundle node may choose to drop lower priority bundles in order to make room for higher priority ones. This error is not propagated back to the source.

* DNS Unreachable: If a bundle agent needs access to its DNS (or DNS-equivalent) and cannot obtain information from it, it generates a DNS Unreachable error. This information is not propagated back to the source application.

* Time Exceeded: If the time-to-live field (either the source-provided TTL or the internal bundle TTL) expires, the source is notified with a Time Exceeded message. These errors are propagated to the source application.

* Source Entity Access Denied: This error indicates that the source

entity does not have access to a needed resource at an IPN node. The source might not be authorized to use the node at all, or it might not have authorization to use a particular interface required by the bundle. Source Entity Access Denied errors

indicate that the source is not authorized to use a particular resource; other errors (e.g. Insufficient Buffer Space) indicate that a particular resource is unavailable to a bundle. For example, an entity on the surface of a planet might be authorized to communicate, using the bundle protocol, with another entity on the other side of the planet via a low-altitude orbiter that is also an IPN gateway. The sender might not, however, be authorized to send bundles across interplanetary space. In this case bundles sent to the orbiter destined for the other side of the planet would not cause errors, while any bundles with off-planet destination addresses would. Source Entity Access Denied errors are propagated back to the source application.

* Invalid Administrative Destination Name: Once a bundle has reached its destination IPN Region, the administrative part of the destination name can be verified. If the administrative part of the destination name is not valid, the source is notified with an Invalid Administrative Destination Name error message. As with LSRR/SSRR, it would probably be advantageous to check the administrative part of the destination name as soon as possible to avoid propagating misnamed bundles and error messages across the backbone.

* Invalid Destination Application: If the destination bundle agent cannot instantiate the destination application (based on the destination application instance handle in the bundle), it notifies the source application with an Invalid Destination Application error message.

* End-to-End Access Denied: If the bundle destination does not accept the bundle due to an authentication or access-control error, the source is notified with an End-to-End Access Denied Message.

4.4. Support of existing Internet applications

There is no clean way to support "legacy" applications in the IPN. One might think that application-layer proxies could protect applications from the rigors of interplanetary communications, but it turns out that this is not the case, even if the appropriate timers (at both application and transport layers) could be scaled correctly. As an example, consider a legacy FTP client on Earth trying to get a file from an FTP server on Mars. If the client's connection request to {foo.bar.com, mars.sol} is somehow coerced into binding to an Earth-resident application proxy (as can be done with nonstandard modifications to the terrestrial DNS) the proxy then has to negotiate enough information out of the client to form the bundle that will

travel to Mars. This bundle needs to include at least:

* The client's IPN domain name
* The server's name tuple on Mars (Note that if wildcarded A-records are used to cause the off-planet server name to bind to the proxy, the proxy will not have a notion of the destination

name. The proxy will have to explicitly request the source name from the source.)
* The file to be retrieved
* Where to deliver the retrieved file to, since it probably won't be coming back in this session

While all of this information could in principle be elicited from a command-line client via appropriate server messages and extensive use of the `quote' command, there is no guarantee that other clients (specifically GUI clients) will be able to accomplish this. SMTP (e-mail) is perhaps the only application that could possibly be tuned to work over interplanetary distances, but even SMTP has embedded timers that would have to be altered significantly before it would work.

5. Security in the IPN

We do not have a detailed list of security requirements for the Interplanetary Internet, primarily because there currently aren't any "users" of the IPN and few, if any, of the potential users have given enough thought to security to commit to a set of security "requirements". However, we know that the Interplanetary Internet's bandwidth resources will be precious. We can also safely assume that the IPN will be a prized hacker's target (at least from Earth). We can also envision that there will be precious, private data flowing across the IPN. As a result, we assume that various security mechanisms and services will be required to provide protection for the bundles traversing the IPN and for the IPN infrastructure itself.

There are two aspects to IPN security: protection of the IPN infrastructure and protection of the data traversing the IPN. The protection of the IPN infrastructure is not unlike the protection required for the Earth-based Internet infrastructure. There will be a need to exchange routing information securely among the IPN nodes as well as to securely manage them. For the IPN infrastructure to protect itself, the IPN nodes need to be mutually suspicious of one another. That is, the IPN nodes will authenticate themselves to each other to ensure that they are not being spoofed by an untrustworthy entity. One might ask if this is overkill if we believe that there will always be a small, controlled number of IPN nodes (a la the original ARPANET). However, one could equally envision that there could be many IPN nodes that are sponsored and controlled by multiple organizations (a la the current Internet). Since we would like the IPN to easily scale, we want to build mutual suspicion and security mechanisms into the IPN architecture from the outset. It should be noted that the same mechanisms could be used to provide security for both the IPN infrastructure and the data flowing through it.

5.1. Assumptions Regarding Required IPN Security Mechanisms

The security mechanisms assumed to be required for the IPN are:

* Access control
* Authentication

* Data integrity
* Data privacy

Access controls will be required because the IPN's space-based assets will have limited resources available and because it will be a likely target from a hacking perspective. By limiting access to the IPN resources, we limit the availability of the IPN to only those users who require its services and do not allow it to be overburdened by those who are not authorized to access the IPN services.

Authentication of identity will be required to perform access control mediation. To allow or disallow access to the IPN, an assured identity of the source of the network traffic will be needed. The identity might be of an individual (e.g., a payload principal investigator) or a location (e.g., a science payload control center or a spacecraft control center).

Data integrity will be required to ensure that data received across the IPN is the same data that was originally sent. This is true from both an IPN infrastructure perspective (e.g., management data) as well as from a user's perspective (e.g., science data, command sequences). Data integrity assures that any unauthorized modification of the data will be detectable by the receiver.

Data privacy will be required to ensure that only those who are authorized to obtain data traversing the IPN or destined to/from the IPN infrastructure will be able to do so. This mechanism is also known as confidentiality.

In the network security community, there are two well-known security paradigms: hop-by-hop security (also known as link security) and end-to-end security. In the hop-by-hop paradigm, the data to be transmitted over the network is protected on a hop-by-hop basis. That is, the data is protected at its source, but in order to be routed to its final destination, it must be unprotected at a trusted routing point (e.g., a ground-based gateway, an IPN gateway) in order to be examined for onward routing. The trusted routing point must then re-protect the data and forward it on to its next routing point. Each successive hop point must unprotect and re-protect the data until it reaches its ultimate destination. A negative aspect of this security paradigm is that, depending on how the protection is applied; the data might be completely exposed while in the gateway (hence the term trusted gateway). This means that the data is potentially vulnerable to unauthorized modification and unauthorized disclosure.

The end-to-end paradigm does not employ trusted gateways. Rather, it

assumes that the path between the source of the data and its destination is hostile and cannot be trusted. As a result, the data is protected at its source and is not unprotected until it reaches its destination. However, in order for this scheme to work, routing information must remain unprotected so that the intermediate gateways

are able to determine how to forward the data without having the opportunity to read or change the data.

One problem with the end-to-end paradigm is that it can only work across a network where there are end-to-end protocols (e.g., TCP). There has to be a protocol below the data that provides the ability to route. One example of this is the Internet Engineering Task Force's (IETF) Internet Protocol Security Protocol (IPSEC). The IPSEC Encapsulating Security Payload (ESP) protocol provides an end-to-end security service between the IP and TCP protocol layers.

However, in the IPN, as has already been discussed earlier, IP and TCP are not necessarily carried end-to-end protocols. Rather, those protocols can/will be terminated in a local internet (e.g. a ground segment on earth or another celestial body) and not carried end-to-end through the IPN. The data will be carried end-to-end via the bundle protocol that would in turn, be transported by other IPN protocols (e.g., LTP, TCP) using the pony express model.

Since the IPN must be structured on a store-and-forward basis, and since users may not trust the IPN gateways, solutions such as IPSEC cannot be employed. In order to provide an end-to-end like security solution, security mechanisms can only be applied to the data and not any other protocol layer(s) below the bundle protocol. This is the way the Secure Sockets Layer (SSL) (now being specified within the IETF as the Transport Layer Security (TLS) protocol) and secure email techniques such as S/MIME and OpenPGP work. In essence, security services are only applied to the data portion of a packet, leaving all of the packet's protocol headers in the open and available for use by intermediate systems. For example, to transmit the string "hello world" across the Internet would entail encapsulating the string into a TCP header, an IP header, and a media access (MAC) layer header. To provide end-to-end security services, only the payload ("hello world") would have security services applied to it (e.g., it might be encrypted to provide privacy/confidentiality). However the TCP, IP, and MAC headers would all remain without security services applied to them. In comparison, when using IPSEC ESP, the TCP header would have security services applied to it to protect it as well as the payload data. And if it were operating in "tunnel" mode, ESP would encapsulate the entire payload - the TCP and IP headers - inside of an IP packet.

Given that data will be transported across the IPN in bundles using an email-like paradigm, borrowing technology from email security is a solution for the IPN.

5.2. Secure Email Technology

Since secure electronic mail operates in a non-interactive mode, and since both communicating parties have not necessarily communicated with one another previously, a technique different than used by IPSEC was developed. One way in which email could be securely wrapped

(e.g., encrypted and/or digitally signed) is via through the use of public key cryptography. Using this technology, an email sender would use the public key of the intended recipient to encrypt the payload (i.e., the message) and the recipient would use its private key to decrypt the payload. Likewise, the sender could digitally sign the message (in order to authenticate the message) using its private key and the recipient would verify the message's authenticity using the sender's public key. However, there are two problems with this scheme for the IPN.

The first problem is that public key cryptography is quite consumptive of processor power. Therefore, the common practice is to use symmetric key (i.e., shared secret) algorithms for large amounts of data. With changes in technology, this problem may disappear, at least for earth-based systems. However, it is not clear that space-based assets will catch up as quickly. The second problem is that public key cryptography is consumptive of bandwidth. A public key exchange technology that allows two communicating entities to derive a shared symmetric key by virtue of knowing each other's public keys and exchanging some random information is known as a Diffie-Hellman exchange. However, the exchange of information must be performed in a near-real-time environment so that the shared key can be used to encrypt transmissions. This is not practical in the IPN since there are no real-time exchanges of data on an end-to-end basis.

However, the secure email community has realized that they also operate under a pony-express model. The secure email community has also realized that secure email may be sent to entities with which one has not previously communicated. Therefore, there needed to be a base-level expectation of a common, minimal set of security services that both sides could use.

The general technique used by the secure email community is that the sending entity decides what security mechanism(s) to employ (e.g., encryption for confidentiality, digital signature for authentication and integrity, or both). If the data to be sent via email is to be encrypted, the sender generates a random key that is used to encrypt the data and the data is encrypted. The sender then either has in its possession the public key of the receiver, or queries a public key server (e.g., a public key infrastructure or PKI) to obtain the receiver's public key, which would be contained in a digital certificate. At the expense of additional bandwidth, the sender can transmit its digital certificate in the email message, which the receiver can verify as genuine based on the well-known certificate authority's signature on the certificate. The key used to encrypt the data is encrypted using the public key of the receiver and the message is sent. The receiver uses its private key to first decrypt

the key used to encrypt the message and then uses the decrypted key to decrypt the data.

When a digital signature is used, the sender calculates a hash of the message using a digest algorithm such as MD5 or the Secure Hash

Algorithm (SHA-1). The resulting hash is then encrypted using the sender's private key. The encrypted hash is sent along with the message data. The receiver uses its copy of the sender's public key to authenticate the fact that the message was sent by the sender.

5.3. Application of Secure Email Technology to the IPN

Given that the IPN and electronic mail share the same operational paradigm, the IPN's notion of "bundle-space" is directly analogous to a secure email MIME body part. As was previously explained, in secure email the contents of the message are encrypted using a symmetric key. The actual message contents are "containerized" (which can be analogized to the contents being "bundled") into a MIME body part. Additional MIME body parts are also attached which would contain the encrypted symmetric key and additional information needed for decryption.

Therefore, the same security techniques can be applied to the IPN's notion of "bundle-space." It can be seen how bundle payload data carried through the IPN is equivalent to an email message. Therefore, the bundle payload data, like the email message, can have security services applied to it before being sent as a protected entity via bundling across the IPN.

In essence, the really difficult part of deploying an IPN security solution based on secure email is the problem of deploying a public key infrastructure (PKI) in the IPN. Deploying a PKI on Earth is equally difficult as can be seen by the continuing problems faced in the various PKI pilot programs currently underway.

On Earth, the problem is how do competing PKIs cross certify public keys and who acts as the root certification authority? (PKI is based on a tree hierarchy where a root certificate authority's public key is well known in order to certify public key certificates). On Earth, when a user does not have the necessary public keys, the user can go off to a public key certificate server to obtain the needed certificates which contain digitally signed (i.e., certified and authenticated) public keys. In the case of email, the sender can attach its public key certificate as a MIME body part. The receiver then validates the digital certificate and uses it to obtain the information contained in the message. Eventually, a cache of digital certificates is formed containing the information needed to securely communicate among a cadre of entities without having to resort to the use of a key/certificate server.

The IPN, however, will not have the luxury of being able to query on-line PKI certificate servers (at least not in the same sense as is

being contemplated on Earth). The problem is very much analogous to the one with the Domain Name System (DNS). When a user sends a secured "bundle" to an IPN entity on another celestial body, we would not want the receiver to have to first query a PKI for a public key certificate on Earth to obtain the receiver's public key. A local

PKI might exist. However there is the issue of the dynamic nature of such a server and the high likelihood that it would quickly drop out of synch. It would seem reasonable that the public key certificates of the various space-based entities would not change often, but this could not be said about the Earth-based entities.

An earth-based sender may be able to query a certificate authority to obtain a certificate for an IPN entity not located on earth. However, an IPN entity in space sending a bundle would not be able to perform such a query. Therefore, it would appear that a set of pre-placed, cached certificates containing the public keys of those IPN entities that are expected to be communicated with would make sense. In addition, an IPN sender should always include its public key certificate as part of the bundled transmission across the IPN as is done using secure email. This would "cost" us in terms of bandwidth (a typical X.509 public key certificate is about 1K bytes in size but can be larger when certificate chains are included). However, by including the certificate, we can be assured that the receiver will not have to use any additional bandwidth, or incur any additional delays in making use of the transmitted data.

5.4. Protecting IPN Data and the IPN Backbone Infrastructure

The IPN will have few and precious resources. Therefore, not only the user data flowing across the IPN will require security services - the backbone infrastructure itself will require similar services. In order for the IPN infrastructure to be self-protecting, it must be built using the paradigm of mutual suspicion. Mutual suspicion requires each entity of the IPN to not assume that another IPN entity with which it is communicating is a "friendly" entity. That is, it should not assume that the IPN entity is who it claims to be without a verified means of identification. Based on a verified identity, access controls can be performed to allow or disallow communications between entities.

Infrastructure information such as routing updates, node management information, distribution of digital certificates, and certificate revocation lists need to be protected from unauthorized modification and potentially from unauthorized disclosure. The IPN nodes would use the same mechanisms as are used to provide protection for user data - namely the security services available to a bundle aware application (e.g., a "bundle-agent").

A bundle aware application would encrypt and/or digitally sign the IPN infrastructure payload to be transmitted to another IPN node. The signed and/or encrypted payload would then be presented to a bundle-agent that would prepare the payload data to be carried across

the IPN in a bundle. Potentially, the bundle-agent might also sign and/or encrypt for hop-by-hop security protection, which would allow each bundle-agent receiving the bundle to authenticate the identity of the transmitting bundle-agent. This mechanism provides the ability to ensure that no other entity can masquerade as a rogue

bundle-agent.

The bundle-agent residing at a ground-based IPN gateway should check the signature of the bundle payload to perform an access control check to limit access to the IPN only to those who are authorized to use it. The access control check can be accomplished via an access control list.

Finally, the receiving application would make the final security checks before accepting the data payload from the final receiving bundle-agent. Should all the final checks succeed, the receiving application would then be free to consume the data.

6. Building a Stable Backbone for the IPN

Just as the performance and capability of the terrestrial Internet are largely determined by the capacity and stability of its backbone links, so will the performance and capability of the Interplanetary Internet depend in large part on the capacity and stability of the interplanetary backbone.

By "backbone" we mean a set of high-capacity, high-availability links between points of access to high-activity subnets. In the terrestrial Internet, backbone links are typically between the high-activity subnets for cities such as Houston and Chicago. In the Interplanetary Internet the backbone links will be between the high-activity "subnets" for planets such as Earth and Mars.

But the IPN backbone will differ from familiar terrestrial backbones in several important ways.

* The media by which information is transmitted obviously differ. Terrestrial backbone links used to be copper wire and are now optical fiber. In the Interplanetary Internet, all information will be via radiation _ either RF or optical _ through empty space.
* The nature of the connectivity between backbone points of presence (POPs) will be fundamentally different. Terrestrial Internet connectivity is structural and relatively static: nodes are physically attached to fiber. Interplanetary Internet connectivity will be operational, directed, and highly dynamic: radiation must be directed at the right moment toward nodes that are prepared to detect it, and the transmitting and receiving entities will be in continuous movement relative to one another.
* The costs of deploying, repairing, and upgrading infrastructure will be much higher for the interplanetary backbone than for terrestrial backbones.
* The costs of configuring, operating, and managing the

interplanetary backbone will likely be somewhat higher than for terrestrial backbones. One factor in this is the scarcity and high cost of electrical power at IPN nodes deployed off Earth.

* Perhaps most important of all, the distances between nodes of the

interplanetary backbone will be vastly greater than the distances between nodes of terrestrial backbones _ so great that the speed of light becomes a very significant constraint on backbone operations.

6.1. Backbone Design Considerations

The differences described above imply two general constraints on the design of the interplanetary backbone.

1. Bandwidth is not free, or even cheap. Reliable delivery cost per bit will be far higher across the interplanetary backbone than across any terrestrial backbone, so bandwidth must be thoughtfully allocated. Both retransmission and forward error correction coding contribute to reliable delivery, but both cost bandwidth; we must seek a balance between the two that minimizes overall overhead.
2. Interactive protocols don't work, at least not well. Negotiation _ connection establishment, flow control, congestion control _ can easily take so long as to consume all available transmission time. Reliable, in-order stream delivery can be so severely delayed by retransmission latency as to be operationally useless.

These design constraints, in turn, must be accommodated at four layers of the protocol stack.

At the physical layer, the relevant research is in radiant RF or optical communication. The physical infrastructure of the interplanetary backbone consists mainly of antennae, many of them mounted aboard orbiting or landed spacecraft, rather than cable in buried or undersea conduit. In the near to medium term, the principal elements of this infrastructure will be Earth-based tracking stations, such as NASA's Deep Space Network, and the planned "Marsnet" of low-Mars orbiters plus a possible areostationary relay satellite. In the long term these assets might be augmented by optical communication satellites orbiting Earth and/or other planetary bodies, and perhaps by additional relay satellites positioned at the LaGrange points of planetary orbits (possibly using solar sailing technology for autonomous station-keeping).

Although this infrastructure will not be subject to some of the problems of terrestrial backbones _ for example, we needn't worry about backhoes cutting through underground fiber _ there will be other challenging operational issues. Accuracy in pointing and transmission scheduling at the backbone antennae will be critical to

assuring the most efficient use of these assets. This means that all elements of the interplanetary backbone infrastructure must share a common understanding of (a) one another's orbital dynamics and (b) the current time. The latter argues for the importance of reliable space-borne clocks, together with a protocol for frequent correction of clock drift based on current navigation data.

Moreover, there will be times at which connectivity between a given pair of backbone antennae will be impossible due to the interposition of large, radiant planetary bodies _ or worse, the sun. These occultations will be predictable, given good models of orbital dynamics, but will nonetheless necessarily constrain the manner in which data flow through the backbone.

At the link layer, the design issues are somewhat less exotic. The interplanetary backbone will require link protocols that minimize overhead and that have no inherent expectations of low noise or low point-to-point transmission latency. CCSDS protocol standards such as Proximity-1 are plausible candidates. They support robust encoding to reduce bit error rates, at some cost in bandwidth consumption.

Our current thinking is that no interplanetary backbone functionality will be required at the network layer. The reason for this is that we expect the endpoints of transport-layer protocol on the backbone to be in direct line-of-sight contact as discussed below; since there will be no intervening nodes to route through, there will be no need for routing functionality at the network layer. [Selection of endpoints for point-to-point transport-layer communication does indeed imply the need for routing and network functionality, but this requirement is addressed at the Bundling layer above Transport.]

Finally, the general constraints on the design of the interplanetary backbone mandate constraints on the protocol used at the transport layer. The TCP transport protocol used in both the backbones and the subnets of the terrestrial Internet _ and of other planetary subnets of the IPN _ will not be suitable: connection negotiation and in-order stream delivery are incompatible with the enormous distances between nodes of the interplanetary backbone.

As noted earlier, the bundle protocol residing just above the transport layer is by nature relatively optimistic about transmission success but it must have transport-layer-like properties, i.e., it must be able to recover from transmission failure at lower layers. The capacity for timeout detection and custodian-to-custodian retransmission has to be built into it.

However, the bundle protocol's structural optimism would result in poor performance if such a capacity were exercised frequently. That optimism has to be justified by the general trustworthiness of lower layers:

* When the bundle protocol runs over TCP in a deployed internet, TCP's own retransmission regime automatically recovers from errors in the network and link layers, and only a failure in TCP itself

will trigger retransmission at the bundle layer.

* When the bundle protocol is operating over interplanetary distances, a similar level of trustworthiness must be provided by a new interplanetary transport protocol.

This protocol will necessarily be very different in operation from TCP. For example, connection time within the interplanetary backbone will be too rare and valuable to squander in waiting for reciprocal traffic of any kind. Consequently the basic mode of operation in the interplanetary transport protocol will be asynchronous: data will be issued continuously throughout each episode of connectivity, with return traffic _ acknowledgements and coordination messages _ matched up to the corresponding original data as it arrives. Data will arrive at the final destination out of order (because retransmitted data will arrive late); in many cases, even out-of-order delivery to applications maybe preferable to waiting the minutes, hours, or even days required for complete, error-free acquisition of data.

Fortunately, out-of-order delivery due to retransmission delay may be made relatively rare by forward error correction at the link layer. Only a failure in that error correction need trigger retransmission at the transport layer.

Accurate timeout detection will be critical to the success of this retransmission regime. Premature timeouts would result in unnecessary retransmission, consuming precious bandwidth and degrading link utilization; late detection of timeouts would delay both bundle delivery and the recovery of retransmission processing and storage resources. We believe interplanetary transmission timeouts can be detected accurately, but only when round-trip times can be calculated from firm operational data rather than estimated from past experience: the extreme variability in round-trip time introduced by intermittent connectivity means that the total round-trip time for transmission N might be hundreds of times greater than that for transmission N _ 1. The required operational data could in theory be provided regardless of the topological relationship between the Transport-layer endpoints, but we believe support for any model other than point-to-point, direct line-of-sight transmission between the endpoints will not scale. [Note that accurate clock synchronization across the interplanetary backbone is as important for implementation of timeouts at the transport layer as it is for resource scheduling at the physical layer.]

Any number of blocks of transport-layer data traffic might concurrently be in various stages of transmission and retransmission, so the aggregate storage space allocated to transmission state data and retransmission buffers might be very large. Moreover, loss of this information due to (for example) an unplanned power cycle could abort any number of transmissions _ unlike an unplanned reboot of a TCP host, which will crash only the messages that are currently being transmitted on all open streams. For these reasons, it may well be

desirable to retain interplanetary transport protocol data in non-volatile storage rather than dynamic memory.

In some cases a set of transport protocol agents _ e.g., the set of relay satellites orbiting a planet _ may be most useful if they can

operate in concert as a single "aggregate" entity. By distributing the retransmission workload across multiple agents as they successively acquire connectivity to a common peer agent, this collective behavior might reduce the total elapsed time required to effect reliable completion of a large block. If so, we'll additionally need some sort of application-layer coordination protocol operating among the members of this aggregate entity.

Flow control and congestion control are possibly the least tractable interplanetary backbone design issues at the transport layer. TCP's strategies are based on reciprocal message exchange that the large delays in interplanetary communication make generally impractical. Alternative approaches remain to be discovered.

7. Deployed Internets in the IPN

We differentiate between two types of networks in the IPN: the long haul backbone and deployed internets. The long haul backbone, described in the previous section, is characterized by long propagation times, due to the great physical separation among the communicating elements. Its protocols are designed to operate efficiently in long-delay, intermittent-connectivity environments. As discussed in [ref: Why not TCP], as delays increase, the efficiency of the Internet suite steadily decreases until applications fail altogether due to embedded time outs.

We designate a network to be a "deployed internet" if it meets the following criteria:

* It has connectivity to an interplanetary gateway, and through that gateway can reach the interplanetary backbone or other deployed networks.
* It has a communication environment that doesn't inherently preclude the use of (possibly enhanced) internet protocols.
* It uses the Domain Name space as a common means of referencing objects and systems across deployed internets.

This designation covers a wide range of possible configurations. The following list provides examples of deployed internets:

* A single lander that hosts an interplanetary gateway and a (real or virtual) lander-internal network;
* A small number of cooperating robots on a planetary surface, such as a single lander and a single rover;
* An orbiter, a lander, and a sample-return vehicle communicating among themselves and via interplanetary gateways hosted in the orbiter and/or the lander;
* Multiple "cells" of robots on a planet surface communicating

beyond line of sight via low-orbiting satellites that serve as relays and as interplanetary gateways;

* Similar scenarios as above, but with one or more planet-stationary satellites serving as interplanetary gateways;

* Spacecraft-onboard networks that communicate with remote endpoints via interplanetary gateways;
* The earth's Internet is the initial instantiation of a "deployed internet."

7.1. Applications of deployed internets in the IPN

It is appropriate to consider the types of activities that will exercise the deployed internets. Clearly, we do not yet know all of the types of applications that will be used in deployed internets, but we do have a basic idea of some of the types of applications that will be used in the relatively near term. In developing this notion of applications within deployed internets, we want to eventually develop the ability to characterize their use of the IPN in terms of arrival rates, data volume, latency and reliability requirements, number and location of producers, number and location of consumers (within the IPN), etc. At this moment, we have only highly qualitative notions of these kinds of data. Over time, we will develop a more quantitative representation of this traffic as an adjunct for testing. The model that we develop will probably never accurately reflect the actual use of the IPN, because the use of the IPN will be affected by its availability: users will adapt their usage patterns as their understanding of the network's capabilities develops. We don't ever expect to hit this moving target, but we hope to come close enough to discover most of the usage-related issues that might exist.

One of the primary uses of a deployed internet is to return science data from the point where it is collected to the point where it will be processed. The processing point could be on a remote internet (say, on earth) or at some point in the local deployed internet. Depending on the resource availability at the gathering site versus the availability at the processing point, the transfer could be largely unprocessed data, shipped as a formatted stream or as a file. Alternatively, the data could be processed to reduce its transmission size, which would likely result in a file transfer operation. Sizes, frequency, precise reliability requirements, and so forth remain to be determined. However, in general, this data is not particularly time-sensitive (although the equipment may be quite sensitive to the amount of time that it takes to complete a transfer due to power considerations, which will be discussed later).

Another primary application, the reporting of health and status telemetry, is typically accomplished via repetitive, unreliable transmission in today's spacecraft. There is underway a slow evolution toward data-summarization on board spacecraft. But there is, understandably, some resistance to filtering or summarizing data

at the spacecraft, since in the event of a spacecraft catastrophe, data not previously transmitted is not available for post-mortem analysis. An important aspect of current telemetry systems is its delivery characteristics, which are either "stream-oriented" or periodically delivered. Mission operators perceive that "heartbeat"-

style information that is ancillary to a periodic or "continuous" stream of telemetry is an important characteristic that will not displace event-driven telemetry for the foreseeable future.

A third type of application is the command and control of in-situ elements. Command and control refers to the closed-loop control of remote systems. The endpoints of the control loop could be separated by interplanetary space, as in the case of terrestrial commanding of a rover. Alternatively, the endpoints could be in reasonably close proximity, but resident on physically separate platforms connected by wireless media. This is the case when a lander controls a rover. These types of applications must be designed to accommodate the delays intrinsic in the network, but the network can permit more responsive control operations by providing qualities of service that mitigate those delays. With command and control applications, data volumes tend to be fairly low, and the commands tend to be followed by responses (although in some instances, command responses are inferred from returned telemetry, described above).

The final type of application we consider is telescience and the development of a "virtual presence." This type of application is intended to synthesize great volumes of information about the local environment in order to allow earth-resident scientists, in-situ controlling robots, or eventually in-situ astronauts to interact with high-fidelity models of a portion of a remote environment. To an extent, this technology has already been used to assist in planning the Mars Pathfinder/Sojourner excursions (ref http://www.piercorp.com/Projects/mars/mars2.htm). However, the technology for fully exploiting this type of adjunct to exploration is still in development. Terrestrial research into telepresence and tele-immersion (ref: http://www.advanced.org/tele-immersion/publications.html) may provide insight into the workload that this type of application might represent for the IPN.

7.2. Characteristics of remote deployed internets in the IPN

Although we consider the Earth's Internet to be the first instance of a deployed internet, the environmental characteristics affecting internets deployed in remote locations may differ significantly from those affecting Earth's Internet. In examining these characteristics, we must differentiate between two different types of environments in the Earth's Internet: the traditional, "wired" environments; and the emerging mobile, ad hoc wireless environments.

First and foremost among the environmental characteristics of note is that of power availability. In the "wired" Internet on Earth, power

is cheap and plentiful. Power availability is more of an issue in terrestrial mobile, ad hoc networks (MANETs), because the mobile nodes operate using portable power sources, typically batteries. However, in terrestrial MANETs, the mobile nodes eventually have access to the same cheap and abundant sources of power that the

"wired" nodes use. This is not the case for remote deployed internets (RDIs) in the IPN. For the foreseeable future, the primary source of power for RDIs in the IPN will be the sun. Solar power conversion is currently relatively inefficient, but even if dramatic improvements in conversion efficiency occur, the fact remains that as one moves away from the sun, the available power diminishes. In the orbit of Mars, the average solar intensity is 590 W/m2, less than half of the 1370 W/m2 in Earth orbit [ref: http://powerweb.lerc.nasa.gov/pv/marspower.html]. On the surface of a planet, the solar intensity is by seasonal variations, by dust build-up on solar panels, and by dust erosion of the solar panels over time. As a result, power availability is and will be of overriding importance to all aspects of communication in RDIs, and will dictate a need for efficiency at all protocol layers.

The signal-to-noise ratios (SNRs) experienced in the terrestrial wired Internet are currently very high, making loss due to data corruption a rare experience. In terrestrial MANETs, SNRs are lower, due to both power availability and to node density. In remote deployed internets, SNRs will be very low, due primarily to power availability.

In the terrestrial wired Internet, the bulk of the routing infrastructure is fixed in terms of its location. Conversely, in terrestrial MANETs, the infrastructure is deployed on an as-needed basis, and may be mobile. In RDIs, this infrastructure is also deployable and mobile, but will have a large component of satellite-based resources.

If we examine the transmission media in use, the terrestrial wired Internet uses primarily copper cable and optical fiber. Terrestrial MANETs use free-space propagation in the radio frequency (RF) range or the infrared range of the electromagnetic spectrum. In RDIs, we anticipate that free-space RF will be the primary communication medium, even for many permanent, immobile installations as they develop, due to the cost of landing, deploying, and maintaining cable or fiber.

The cost characteristics of deployment, operations, repair of RDIs is not currently well understood. However, deployment cost of anything that must be landed on the surface of another planetary body is quite high. Correspondingly, repair or replacement costs of components of the RDIs are also high. We do not yet have a clear understanding of how the operations costs of the RDIs will compare to the operating costs of terrestrial MANETs or of the terrestrial wired Internet.

7.3. Effects of environmental characteristics on protocols for the

IPN RDIs

In considering how to deal with the anticipated operating environment for RDIs, we are encouraged that a significant amount of relevant research is currently under way to support protocols for terrestrial

MANETs. This section briefly reviews some of the areas in which work is required, and is organized by protocol layer.

At the physical layer, spectrum management is an issue that requires coordination. There is currently no "Federal Communications Commission" or International Telecommunications Union that regulates use of radio-frequency spectrum outside of earth. However, coordination of this sort is needed sooner rather than later, due to the attractiveness of certain frequency bands as a result of their free-space propagation loss and refractive characteristics [ref: private communication].

The cost of landing equipment on remote planetary surfaces motivates designers to keep as much communication infrastructure in orbit as possible. Low-orbiting satellites will be used extensively to support surface-to-surface communication and RDI-to-RDI communication. Antenna design to support wideband communication between surface-based mobile nodes and these low-orbiting satellites is an area where research might yield great benefit. Although we have heard of some research in these areas to support military on-the-move applications, we have seen nothing in detail and feel that it is an area that is ripe for additional study.

At the link layer, management of low SNRs is of significant importance, with many areas of relevant research. While many different coding schemes are available, it is not yet clear whether one of these is best for all RDI use or whether the QOS requirements of different types of traffic will dictate the use of different coding schemes on a per-packet basis. Many codes are currently being considered, including convolutional coding, concatenated codes (such as Reed Solomon codes), and the emerging Turbo codes. Each of these has different properties related to delay, residual error rate, and link acquisition characteristics.

Resource reservation schemes at the media access level may be of significant importance in supporting closed-loop control operations. Some of these schemes have the benefits of providing bounded delays and of avoiding interference, which is important in power-constrained environments.

Link-layer status detection and signaling (to upper layers) is important in resource-constrained environments. We believe that the link layers must detect and signal link availability, link capacity and congestion status, and current error conditions. As the terrestrial cellular telephone industry becomes internet-enabled, some of these issues are beginning to be addressed in single-hop wireless environments. There is also ongoing research of interest in

MANETs and in DARPA-sponsored research [ref. ftp://ftp.rooftop.com/pub/apis/link_api.pdf].

At the network layer, there is a need for routing protocols to support both fast- and slow-moving mobile nodes. The routing

protocols must be able to constitute and maintain networks comprised of a combination of fixed and mobile nodes. Significant research in this area is currently being performed, and is being coordinated by the IETF's MANET working group [ref. http://www.ietf.org/html.charters/manet-charter.html].

Another area of relevant network-layer research concerns "vertical handoffs," which allow nodes to adapt to changes in available links or the resources available via particular links. This work [ref. http://daedalus.cs.Berkeley.edu/publications/monet97.ps.gz] is focused on "last-hop" use. Its potential for use within wireless routers requires further research.

As link layers enhance their capabilities to monitor and report their status internally to the network layer, network layer control protocols and algorithms must be enhanced to appropriately propagate this information to affected parties within the RDI. This topic is an active research area, with some applicable work having been done within the SCPS project [ref: http://www.scps.org].

Just as resource reservation schemes within the link layer may be important mechanisms for providing bounded link-layer delays, resource allocation schemes at the network layer may also be important for providing such bounds on end-to-end communication within the RDI. However, resource allocation and resource reservation within mobile wireless networks is still a subject of ongoing research. Should wireless networks use the integrated services model of resource allocation? Or should they use the differentiated services model? It is relatively clear that guarantees of network capacity are not feasible in a highly dynamic environment. However, the specification of operating ranges may be more tractable, and may provide useful services

* To applications (by giving them feedback on available network capacity),
* To the transport layer (by providing dynamic rate control information), and
* To the network layer (by ensuring that resources are not over allocated in the non-bottleneck portions of the network).

The applicability to MANETs of the current integrated services versus differentiated services models is the subject of ongoing research [ref: http://comet.ctr.columbia.edu/insignia/, http://www.cs.ucla.edu/~terzis/mobile.html].

The networks that will be deployed remotely are truly without any fixed infrastructure. Therefore, the elements of this network will need to establish and maintain the set of services necessary to

bootstrap the network. Self-configuration has relevance here in the areas of addresses allocation and management, name-to-address binding, and dynamic hierarchical organization. There is ongoing research in all of these areas that is of potential utility, ref: http://www.ietf.org/html.charters/zeroconf-charter.html

http://www.ietf.org/html.charters/dhc-charter.html
http://www.ietf.org/html.charters/svrloc-charter.html
http://www.ietf.org/html.charters/dnsind-charter.html
http://gershwin.utdallas.edu/publications].

At the transport layer, there is a need for development of new protocols or extensions to existing protocols that are capable of participating in power efficient and mobility-aware communication schemes [ref: http://www.isi.edu/workshop/public_html/wmcw97/nsf4.html].

These enhanced transport protocols must be able to adapt to changing network conditions. This applies in particular to adaptation of the protocol's behavior in order to meet application Quality of Service requirements. Additionally, support is required for explicit signaling of and responses to changing network conditions such as link outages, congestion, and corruption trends.

For the foreseeable future, some links in the RDIs will exhibit significant data rate asymmetry (on the order of 100s: 1). The effective asymmetries may be even higher, since the low-capacity link may not necessarily be dedicated to supporting the movement of data across the high capacity link. Accordingly, accommodation of significant data rate asymmetries will be required while still maintaining a high degree of power efficiency in the presence of errors.

At the application layer, service location in mobile ad hoc networks is an issue that is closely related to self-configuration at the network layer, and is of current research interest, ref: http://www.ietf.org/html.charters/zeroconf-charter.html, http://www.ietf.org/html.charters/svrloc-charter.html). This is important both in initial network startup, and also in the (potentially frequent) case that RDIs with low connectivity become partitioned, either expectedly or unexpectedly.

Efficient, autonomous network management and control in RDIs is an area of interest. There is some ongoing research in the area, ref: http://www.ietf.org/html.charters/disman-charter.html, ftp://ftp.ece.orst.edu:/pub/users/singh/papers/anmp.ps.gz that is of potential interest, but this is an area in which further research is required.

Finally, monitoring the health and status of mobile nodes (not necessarily just the networking components) is of significant importance in RDIs. The cost to land new elements on a planetary surface is extremely high. Therefore, we are motivated to perform

preemptive maintenance and to repair, rather than replace, failed parts. The ability to perform these activities autonomously is an area where a significant amount of research is required (and sounds really fun).

7.4. Summary

The effects of power limitations in RDIs are significant and will be present in at least some portions of the IPN for the foreseeable future. These effects strongly suggest the likelihood of some divergence from the standard internet suite of protocols that is in current use on Earth. Whether these changes become the standard internet suite on Earth, as a result of the significant increase in the use of mobile, wireless internet nodes, remains to be determined. It is also likely that some of the RDIs that are deployed will use a model of organization that is substantially different than that employed in the Internet. It may be that the IPN treats these networks as single, distributed nodes, but further investigation is required.

8. Working Conclusions

With the increasing pace of space exploration, Earth will distribute large numbers of robotic vehicles, landers, and possibly even humans, to asteroids and other planets in the coming decades. Possible future missions include lander/rover/orbiter sets, sample return missions, aircraft communicating with orbiters, and outposts of humans or computers remotely operating rovers. All of these missions involve clusters of entities in relatively close proximity communicating with each other in challenging environments. These clusters, in turn, will be in intermittent contact with one another, possibly across interplanetary space. This dual-mode communications environment: relatively cheap round-trips and more constant connectivity when communicating with "local" elements coupled with the very long-delay and intermittent connectivity with non-local elements has led us to the architecture just described, with the following working conclusions.

We need to use a "Pony-Express" model and bundles for interplanetary communication

For communicating between IPN domains such as Earth and Mars, the standard Internet protocols fail, both at the application and transport layers, mainly due to the large delays. Intermittent connectivity and bit error rate are also significant issues, but delay cannot be mitigated by any current means. Communicating over these distances requires a fundamentally different model that does not assume that round-trip-times are cheap. To combat this we propose a "pony-express" model, where senders submit bundles describing entire transactions to the network, which then uses custody transfers to move bundles from node to node. Under this model the sender has little or no expectation of real-time delivery

of the bundle, which may be stored at intermediate nodes for arbitrary periods of time. In time, the communications links connecting the various deployed internets will grow to form a stable interplanetary backbone network.

We can't support most legacy applications, even with application-layer proxies

> SMTP (e-mail) is perhaps the only application that could possibly be tuned to work over interplanetary distances, but even SMTP has embedded timers that would have to be altered significantly before it would work.

Name tuples consisting of a routing part and an administrative part are the means of reference

> To assist in interplanetary routing, we have introduced the notion of an IPN name tuple. These tuples, each consisting of a topology-related routing part and an administratively controlled administrative part are the analog of IP addresses in the terrestrial Internet. Just as with IP addresses, bundles carry these name tuples (source and destination) end-to-end. The routing part of an IPN name tuple serves as an identifier of the destination internet and is consulted to find the next bundle-hop destination whenever the bundle is NOT in its destination IPN region. The administrative part of an IPN name tuple serves the same function as DNS names in the terrestrial Internet, allowing symbolic names to be used in place of network-layer addresses. For the IPN region that includes Earth, and possibly others, we envision using actual DNS names as the administrative parts of the tuples.
>
> This two-part naming approach has a number of advantages. First, it explicitly allows routing information to be encoded in the IPN name tuple (in the routing part) without interfering with the administrative part. Second, since the administrative parts of name tuples are only resolved in the destination IPN region, administrative parts of IPN names do not have to be exchanged between IPN regions for the purpose of routing. Third, while all IPN nodes must be able to interpret the routing part of a name tuple, different name-to-address binding mechanisms for the administrative part can be used in the different regions. Finally, decoupling the administrative parts of the various IPN regions from one another allows autonomy of the administrative naming in different regions.

IPN Nodes will be used as "impedance-matchers" between the (relatively) rich environment of local communications and the long-delay interplanetary environment

> We refer to the nodes that perform the store-and-forward operations on bundles simply as interplanetary nodes, and the bundle-agents on these nodes are responsible for routing bundles through the IPN.

We need flexible and bandwidth-efficient security (particularly

authentication and access control)

   For the foreseeable future, the Interplanetary Internet will be an
   expensive resource, and an irresistible lure to hackers. For these

reasons, access to the IPN will need to be strictly controlled and IPN gateways and resources will need to authenticate commands sent to them. In many cases, data privacy will be required by scientists to ensure that they get first access to the data from their instruments, and integrity will of course be required for command sequences and some telemetry return. At the same time, the goal of the IPN is to allow scientists, researchers, and on occasion the general public, easy access to information from outside of Earth's domain. These competing goals will require a flexible security scheme.

In addition to being flexible, the security mechanisms used over the deep-space links will also need to be bit efficient. Standard Diffie-Hellman exchanges cannot be used to generate traffic keys, as they require the exchange of several rounds of random information in near-real time. Something similar to secure email may be a better approach, where the sender applies the appropriate security to the payload of a bundle and then attaches its public key certificate to the bundle. This approach, while costly in terms of bandwidth, will allow the receiver to immediately authenticate/verify/decode the contents of received bundles.

The long-haul transport protocol used to carry bundles in interplanetary space will be very different from TCP

The transport layer that moves these bundles across interplanetary space will necessarily be very different from TCP, the predominant transport protocol in the terrestrial Internet. Since bandwidth is a precious commodity in the IPN, the LTP must be "connectionless" in that it cannot wait a round-trip to establish a connection before beginning to send data. It is quite possible that the LTP will provide partial data delivery where data with "holes" or errors is delivered to the application out of order instead of in order, as with TCP. A major challenge of the LTP is how to perform flow and congestion control without timely feedback.

Deployed internets may or may not use internet protocols.

For the "local" communications, between nodes on a spacecraft LAN, elements on the surface of a planet, or between elements on the surface and in orbit, for example, round-trip times are comparable to those in the terrestrial Internet. In these cases, it is feasible and indeed very reasonable to use protocols like the standard Internet Protocols (TCP/IP). Using an internet model, and perhaps even versions of the standard internet protocols themselves, will allow deployed devices to easily communicate with one another and to share a common communications infrastructure. A common, shared infrastructure will free each mission from having to design, build,

test, and operate its own custom communications system, and will pay off in terms of development time, development cost, mass, interoperability (reliability), and efficiency. Thus we envision the deployments of fragments that are similar to Earth's Internet to remote locations of interest. Current advances such as Mobile IP and

SCPS that extend the internet model to include mobile nodes and stressed environments will be useful in these environments.

At the same time, it may be advantageous to use protocols entirely foreign to the internet suite in an "RDI." For example, it might be preferable in some instances to deploy a self-organizing sensor network in which only data sets, not individual nodes, are addressable. We believe that the naming method under consideration, where each entity name is divided into an IPN domain part and a domain-specific part, would support these types of RDIs. The mechanics of how to interface such a network with the rest of the IPN and whether/how such a network could be integrated with an RDI using internet protocols are topics of current discussion.

Terrestrial advances in Mobile Ad Hoc networking are necessary but possibly not sufficient for RDIs

There is a large and growing body of ongoing work in the IETF, universities, and government and private agencies to develop protocols for mobile ad hoc networks. While much of this work will be directly applicable to the remotely deployed internets, RDIs will remain a breed apart from terrestrial wireless networks. The main differences are in power and node density. Terrestrial MANETs, while they may be power-starved, still have the prospect of recharging at a wall socket; elements of an RDI have no such expectation. Secondly, the density of elements in an RDI may be low compared to a typical terrestrial MANET (although it is too soon to know that a "typical" terrestrial MANET will look like). The node density may be so low in fact that real-time communications between members of the RDI may not be possible. This will be the case if many landed elements communicate with one another via a small number of low-altitude orbiters, for example.

9. Security Considerations

Security is an integral concern of the Interplanetary Internet. Section 5 of this document is devoted to examining the security requirements in the IPN, some potential approaches for securing data in the IPN, and some approaches for securing the backbone infrastructure of the IPN.

10. Authors' Addresses

Dr. Vinton G. Cerf
MCI WorldCom
22001 Loudoun County Parkway
Building F2, Room 4115, ATTN: Vint Cerf
Ashburn, VA 20147
Telephone +1 (703) 886-1690
FAX +1 (703) 886-0047
Email vcerf@mci.net

Scott C. Burleigh
Jet Propulsion Laboratory
4800 Oak Grove Drive
M/S: 179-206
Pasadena, CA 91109-8099
Telephone +1 (818) 393-3353
FAX +1 (818) 354-1075
Email Scott.Burleigh@jpl.nasa.gov

Adrian J. Hooke
Jet Propulsion Laboratory
4800 Oak Grove Drive
M/S: 303-400
Pasadena, CA 91109-8099
Telephone +1 (818) 354-3063
FAX +1 (818) 393-3575
Email Adrian.Hooke@jpl.nasa.gov

Leigh Torgerson
Jet Propulsion Laboratory
4800 Oak Grove Drive
M/S: T1710-
Pasadena, CA 91109-8099
Telephone +1 (818) 393-0695
FAX +1 (818) 354-9068
Email Leigh.Torgerson@jpl.nasa.gov

Robert C. Durst
The MITRE Corporation
1820 Dolley Madison Blvd.
M/S W650
McLean, VA 22102
Telephone +1 (703) 883-7535
FAX +1 (703) 883-7142
Email durst@mitre.org

Dr. Keith L. Scott
The MITRE Corporation
1820 Dolley Madison Blvd.
M/S W650
McLean, VA 22102
Telephone +1 (703) 883-6547
FAX +1 (703) 883-7142
Email kscott@mitre.org

Eric J. Travis
Global Science and Technology, Inc.
6411 Ivy Lane
Suite 300
Greenbelt, MD 20770
Telephone +1 (301) 474-9696
FAX +1 (301) 474-5970
Email travis@gst.com

Howard S. Weiss
SPARTA, Inc.
9861 Broken Land Parkway
Columbia, MD 21046
Telephone +1 (410) 381-9400 x201
FAX +1 (410) 381-5559
Email hsw@columbia.sparta.com