A DTN Testbed Architecture

Efthymios Koutsogiannis
Department of Electrical and Computer Engineering
Democritus University of Thrace
Xanthi, GREECE
ekoutsog@ee.duth.gr

Sotirios Diamantopoulos
Department of Electrical and Computer Engineering
Democritus University of Thrace
Xanthi, GREECE
sdiaman@ee.duth.gr

Vassilis Tsaoussidis
Department of Electrical and Computer Engineering
Democritus University of Thrace
Xanthi, GREECE
vtsaousi@ee.duth.gr

***Abstract*—As the space communications infrastructure of the near future is yet to be standardized, many protocols have been proposed to cope with issues such as long propagation delays, high packet error rates and intermittent connectivity. In that context, we have designed a testbed, which incorporates Space conditions, mission scenarios and Space communication requirements, rendering the extensive evaluation and design of space protocols feasible and efficient.**

Keywords–Delay-Tolerant Networking; testbed; ION; CFDP; deep-space communications

I. INTRODUCTION

Delay Tolerant Networking (DTN) [1] constitutes an emerging network architecture that facilitates data transfers in challenged networks, characterized by intermittent connectivity, high loss rates and long propagation delays. Although extensive research has been conducted regarding the design and implementation of DTN-specific protocols and protocol stacks, focus is mainly given on the Bundle Protocol (BP) [2]. The latter is an application layer protocol that supports custody-based retransmission, late binding and is able to cope with intermittent connectivity, taking advantage of scheduled, predicted and opportunistic contacts. Nevertheless, BP functionality lacks several routing and reliable transfer features, therefore other protocols need to be deployed as well, in order to provide an efficient solution. In the context of our project "Extending Internet into Space – ESA/ESOC DTN/IP Testbed Implementation and Evaluation" [3] funded by European Space Agency (ESA), we have deployed a DTN testbed for space communications and intend to design and evaluate space-suitable DTN protocols, architectures and routing policies to allow efficient deep-space communications.

II. TESTBED DESCRIPTION

A. *Design Goals*

The main objectives of the DTN testbed regarding its accuracy and efficiency are:

(i) Dynamic control of network parameters. The testbed should be able to emulate fundamental network parameters (such as bandwidth, packet error rate, propagation delay, and available connectivity), and adapt realistically and dynamically to changes in those parameters in real-time.

(ii) Scalability. Although current deep-space communications involve a limited number of communication nodes, the testbed should be able to scale well over a larger number of communication nodes to allow for emulation of any future deep-space communication scenarios, which will include several planetary surface networks and relay satellites.

(iii) Transparency. Network emulation should be transparent to upper layer protocols and applications, thus permitting their use and evaluation without need for modification.

(iv) Flexibility. The testbed should be flexible enough to: emulate any space communication topology; incorporate new protocols, applications and mechanisms; interoperate with other similar DTN testbeds; and provide a reusable infrastructure towards an actual hardware testbed.

B. *Architecture and Topology*

Testbed architecture is designed to allow for clear distinction of the components and facilitate any upgrade (Figure 1). Graphical User Interface (GUI), Central Management System (CMS) and Kinematics Modeling System (KMS) compose the administrative part of the testbed. The user can set the experiment parameters via the GUI regarding the nodes (number of participating nodes, data consumption – production, available storage size), the links (bandwidth, error rate, propagation delay) which can also be modified as the experiment is on progress, and the available protocols. Moreover the GUI is also responsible for the presentation of testbed statistics and status information in real-time.

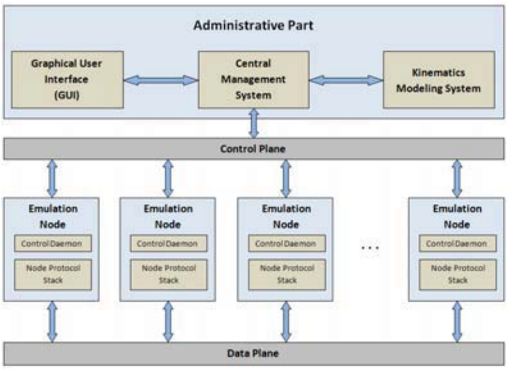

Figure 1. Testbed architecture

User data is being processed by KMS in order to create the deep-space communication scenario and create the corresponding control data for the nodes. CMS is the main component of the administrative part, as it handles the communication between the other components and the exchange of control data with the Emulation Nodes (EN). CMS communicates with ENs and more specifically their Control Daemon (CD) component through the Control plane in order to set node parameters and receive status reports.

ENs are individual PCs supplied with the Node Protocol Stack under evaluation and the necessary Control Daemon software for the communication with the administrative part. Communication amongst ENs takes place through the Data plane exclusively, exchanging files such as images, measurements etc.

Currently the testbed consists of more than ten nodes. An intercontinental link – node (Boston - USA) is available, in order to study the behavior of DTN in terrestrial scenarios under realistic conditions. Furthermore a real geostationary link through Hellas Sat GEO Satellite is integrated into the testbed in order to study low-orbit scenarios.

C. *Protocol Stack*

Interplanetary Overlay Network (ION) [4] is the DTN architecture implementation we use in our testbed. ION is an implementation of (i) the Bundle Protocol, which is used as an overlay network protocol, (ii) the Asynchronous Message Service (AMS), (iii) the Licklider Transmission Protocol (LTP) [5] which provides retransmission-based reliability, (iv) Contact Graph Routing (CGR) [6] which enables dynamic routing. It also supports UDP as an LTP convergence layer and TCP. The UDP convergence layer interoperates with DTN2 [7], a reference Bundle Protocol implementation.

In order to have a more efficient application layer protocol, we integrated CCSDS File Delivery Protocol (CFDP) [8] into the testbed. CFDP allows an automatic, reliable file transfer between spacecraft and ground, designed to support the operation of spacecraft by means of file transfer and remote file system management. It also provides file segmentation for effective transmission, advanced file operations such as remote filestore management commands synchronized with successful file delivery and remote directory listing.

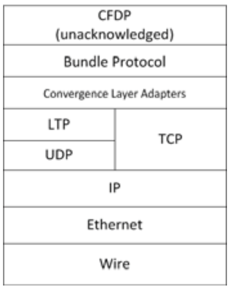

Figure 2. Testbed protocol stack

The integration of CFDP with ION was feasible due to the implementation of a middleware that handles the communication between them. The middleware encapsulates CFDP PDUs to bundles and translates the addressing scheme of CFDP to the corresponding of ION. ION handles the transmission and routing of the data, while CFDP is used as an end-to-end application. The initial evaluation results [9] highlighted the advantages of this integration in conjunction with aspects that need improvement.

III. FUTURE WORK

In the following months we plan to develop a reliable transport protocol borrowing functionality features from both Deep Space Transport Protocol (DS-TP) [10] and Delay-Tolerant Transport Protocol (DTTP) [11]. The new protocol will combine proactive transmission and rate-based transmission behavior, as well as parallel data transfer exploiting various communication opportunities.

One of our main goals is also to implement an efficient routing scheme and integrate it into the testbed. We have already evaluated the performance of several protocols such as PRoPHET [12], CGR [13] and we intend to use the results in order to create an efficient routing protocol.

We plan to implement an advanced Kinematics Module that will automatically adjust link characteristics according to deep space conditions, based on planet movement, random solar storms, etc. The new KM will allow for more realistic emulation conditions resulting in a more efficient evaluation of the protocols.

REFERENCES

[1] V. Cerf, et al, Delay-Tolerant Network Architecture, IETF RFC 4838, Internet Engineering Task Force, April 2007 URL: http://www.ietf.org/rfc/rfc4838.txt

[2] K. Scott, S. Burleigh, Bundle Protocol Specification, IETF RFC 5050, Internet Engineering Task Force, November 2007 URL: http://www.ietf.org/rfc/rfc5050.txt

[3] C. Samaras, I. Komnios, S. Diamantopoulos, E. Koutsogiannis, V. Tsaoussidis, G. Papastergiou and N. Peccia, "Extending Internet Into Space - ESA DTN Testbed Implementation and Evaluation", MOBILIGHT 2009, Workshop on Research activities funded by ESA in Greece, Athens, Greece, May 18-20, 2009.

[4] Jet Propulsion Laboratory, "ION: Interplanetary Overlay Network", URL: https://ion.ocp.ohiou.edu/

[5] M. Ramadas, S. Burleigh, and S. Farrell, "Licklider Transmission Protocol – Specification", IETF RFC 5326, Experimental, September 2008, URL: http://www.ietf.org/rfc/rfc5326.txt

[6] S. Burleigh, "Dynamic Routing for Delay-Tolerant Networking in Space Flight Operations", SpaceOps 2008 Conference.

[7] DTN2 Reference Implementation, URL: http://www.dtnrg.org/

[8] "CCSDS File Delivery Protocol (CFDP) Part 1: Introduction and Overview", Informational Report, CCSDS 720.1-G-3. Green Book. Issue 3. April 2007.

[9] E. Koutsogiannis, G. Papastergiou and V. Tsaoussidis, "Evaluation of CCSD File Delivery Protocol over Delay Tolerant Networks", IWSSC 2009, Siena, Italy, 2009, to appear.

[10] Giorgos Papastergiou, Ioannis Psaras and Vassilis Tsaoussidis "Deep-Space Transport Protocol: A Novel Transport Scheme for Space DTNs" Computer Communications (COMCOM), Elsevier Science, Special Issue on Delay-/Disruption-Tolerant Networks, in press.

[11] Christos V. Samaras and V. Tsaoussidis, "DTTP: A Delay-Tolerant Transport Protocol for Space Internetworks", 2nd ERCIM Workshop on eMobility, Tampere, Finland, May, 2008.

[12] Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) Internet Draft, URL: http://www.dtnrg.org/docs/specs/draft-lindgren-dtnrg-prophet-02.txt

[13] I. Komnios, S. Diamantopoulos and V. Tsaoussidis, "Evaluation of Dynamic DTN Routing Protocols in Space Environment", IWSSC 2009, Siena, Italy, 2009, to appear.