Abstract Submission for Space Internet Workshop  
June 4-5, 2003 - Cleveland, Ohio

Author: David Andrew Carek, P.E.  
Affiliation: NASA Glenn Research Center, Satellite Networks and Architectures  
Address: M.S. 54-5  
21000 Brookpark Road  
Brookpark, OH 44135  
Phone: (216) 433-8396  
Email: David.A.Carek@nasa.gov

Title: Shuttle Payload Ground Command and Control – An Experiment Implementation for STS-107

Abstract:

This presentation covers the design of a command and control architecture developed by the author for the Combustion Module-2 microgravity experiment, which flew aboard the STS-107 Shuttle mission. The design was implemented to satisfy a hybrid network that utilized TCP/IP for both the onboard segment and ground segment, with an intermediary unreliable transport for the space to ground segment.

With the infusion of Internet networking technologies into Space Shuttle, Space Station, and spacecraft avionics systems, comes the need for robust methodologies for ground command and control. Considerations of high bit error links, and unreliable transport over intermittent links must be considered in such systems. Internet protocols applied to these systems, coupled with the appropriate application layer protections, can provide adequate communication architectures for command and control. However, there are inherent limitations and additional complexities added by the use of Internet protocols that must be considered during the design.

This presentation will discuss the rationale for the framework and protocol algorithms developed by the author. A summary of design considerations, implantation issues, and lessons learned will be presented. A summary of mission results using this communications architecture will be presented. Additionally, areas of further needed investigation will be identified.
Shuttle Payload Ground Command and Control
An Experiment Implementation
Combustion Module-2 Software Development, STS-107

Presented at the Space Internet Workshop III
Cleveland, Ohio
June 4, 2003
Combustion Module-2
Ground Command & Control

- System Overview
- Design Considerations
- Protocol Overview
- Implantation Issues
- Lessons Learned
- Summary of Mission Results
- Areas for Further Investigation
CM-2 Software Team

Top Row: Jonancy Colbrunn, Alan Richard, Steve Lux, Kevin Carmichael, Laura Maynard-Nelson, Lisa VanderAar, Dan Taylor
Bottom Row: Len Marinis, David Carek, Jeff Spiegler

Not Pictured: Richard Woodward, Kin Wong, Laszlo Szijarto

Glenn Research Center
David Andrew Carek, P.E.

at Lewis Field
System Overview
SPACEHAB Carrier

STS-107 SPACEHAB Payload
Module Integration - Cape Canaveral

CM-2 installed in the SPACEHAB module.

SPACEHAB Research Double Module (RDM, 17 ft.)

Glenn Research Center
David Andrew Carek, P.E.

at Lewis Field
System Overview
Microgravity Combustion Experiments

LSP

Mist

SOFBALL

See Ref. 1 for further information on these experiments

Glenn Research Center
David Andrew Carek, P.E.
at Lewis Field

NASA
System Overview
Communications Interfaces SpaceLab/SPACEHAB

- **SpaceLab - STS-83, STS-94**
  - Main Computer I/F – RAU (Remote Acquisition Unit)
  - Uplink Test Parameters Only

- **SPACEHAB – STS-107**
  - Main Computer I/F – Ethernet
  - Full Ground Commanding
  - File Downlink
System Overview
End to End Communications Path

Custom UDP like Protocol

SpaceHab Flight Data System  uplink  SpaceHab Ground Data System
downlink

Orbiter Data System
TDRSS - Whitesands
DomSat – JSC PCC

CM-2 Payload

TCP/IP Ethernet

Server

CM-2 Ground Command Computer

TCP/IP Ethernet

Client

Inset graphic source: Ref. 2

Glenn Research Center
David Andrew Carek, P.E.

at Lewis Field

NASA
System Overview
CM-2 Payload Communications Interfaces

- VCR Package
- Image Processor
- Experiment Mounting Structure
- Main Computer
- Crew Control via PGSC Laptop (picture STS-94)
- RS422
- Discrete/Analog I/O
- RS232
- Gas Chromatograph
- Ethernet
- HRM I/F
- EIU
- SPACEHAB Ethernet
- CM-2 Ground Control

Glenn Research Center
David Andrew Carek, P.E.

at Lewis Field

NASA
Communications Design Considerations

• Command Protocol Requirements
  – Flight computer must verify command integrity
    • No end-to-end TCP data integrity
    • Even if end-to-end, TCP checksum not good enough (Ref. 3,4)
  – Ground computer must verify command was processed
  – Must handle arbitrary data errors
    • Multiple bit errors
    • Partial/complete packet loss
    • Multiple consecutive packet loss
    • Long term or intermittent LOS
  – Must work for both PGSC (onboard) and ground commanding
Generic Command Verification Protocol Overview

1. Command Sender generates unique CRC signature by appending timestamp to command prior to CRC calculation.
2. Command Sender transmits command to Command Receiver.
3. Command Receiver verifies command with CRC.
4. CRC signature is statically inserted into the Health & Status data frame.
5. Health & Status frame downlinked synchronously at 1 Hz.
6. Command Sender waits to see CRC signature in H&S frame to verify command was processed.

**Although depicted with ground command source, this is the same protocol that is used for onboard commanding with the PGSC over RS232.**
Uplink Command Frame

• Uplink Command Frame Format

<table>
<thead>
<tr>
<th>Cmd Type</th>
<th>Cmd ID</th>
<th>Data</th>
<th>Timestamp</th>
<th>32 bit CRC</th>
<th>CR</th>
</tr>
</thead>
</table>

- Command ID and command type both 16 bit
- Data: optional and variable size (used mostly to uplink test parameters)
- Timestamp 32 bit (milliseconds of uptime; wraps after 49 days)
  - Timestamp guarantees unique CRC signature for every command
  - Allows discrimination of duplicate command transmission
  - Alternatively a small (1byte) wrapping sequence counter could be used

• Command Encoding

- Command generated in binary (including CRC)
- Converted to hexadecimal ASCII string (0-9,A-F)
- Carriage Return Record Terminator Added to hex string
  - Allows receiver to use simple readln call
  - Facilitates debugging (ASCII hex messages printable)
  - But ... uses double the data bandwidth
- Typically small command message size (nominally 25 bytes up to 256 bytes)
Downlink Data Frame

- Health & Status Engineering Frame Format
  - Binary to ground (300 bytes/sec)

<table>
<thead>
<tr>
<th>Sync Word 0xFFFFFFFF</th>
<th>Frame ID</th>
<th>TimeStamp</th>
<th>Data</th>
<th>CRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cmd ID</td>
<td>Calculated Cmd CRC</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- 32 bit sync word – picked least likely value to occur in data
  - CM-2 mostly 12 bit A/D’s with upper 4 bits unused.
    Therefore 0xFFFFFFFF was an unlikely value to occur and
    good choice for sync pattern (could add alternating bit if
    potential for continuous stream of 0xFF)
Implementation Issues

• Errors and Incompatibilities between TCP stacks
  – System crash on non-TCP packet reception
  – Problem with reconnects

• Data Packet boundaries not preserved
  – Stream preserved but packets arbitrarily fragmented
  – Parser initially designed for aggregation but not fragmentation (frames designed to fit within 1 MTU)
  – Required indexing search algorithm on CM data server (good practice anyway – TCP provides no record marker)

• Large queuing delay
Lessons Learned

• Use of TCP/IP in flight system
  – Doesn’t relax testing requirements
    • sometimes makes problems more difficult to resolve due to lack of source code
  – Standards not implemented uniformly
  – RFC’s hard to use as interface specifications
    • Significant cross referencing
    • Contains much more than may be used/needed by the system
  – Greatly enhances software development and testing

• Full integrated system tests required
  – Only tested on a per payload basis
  – SPACEHAB data system required patch during mission
Summary of Mission Results
Analysis of Health & Status Downlink

Number of consecutive dropped 1Hz frames
Sample of combined AOS & LOS data
Average 18-20% drop rate

Consecutive Dropped Frames

Consecutive Dropped Frames

1/19 1/21 1/23 1/25 1/27 1/29 1/31

Glenn Research Center
David Andrew Carek, P.E.

at Lewis Field
Summary of Mission Results

• High portion of Science data downlinked
  - LSP - 50%
  - SOFBALL - 65%
  - Water Mist - 90%

• Ground Commanding Results
  - No bad CRC’s detected for commanding
    • SPACEHAB up/downlink included checksum
      - Bad data discarded (not forwarded to payload)
  - Commands uplinked = 1228
  - Unique CRC’s in downlink = 1067
  - Average Round Trip Time = 10 seconds
Areas for Further Investigation

• Existing Architecture
  – Quality of service provisions
    • Examine use of TCP to throttle transmission rate
    • Examine impact to real-time OS
  – Worst case S-band latency not fully understood
  – Reduce queuing latency over S-band

• Revised Architecture
  – End to end design using UDP
  – Examine SCPS-TP or gateway architecture (ref. 5)
References

1. NASA GRC, http://microgravity.grc.nasa.gov/combustion/cm/cm_index.htm
3. Boeing, “SPACEHAB Experiment Interface Definition Document” MDC91W5023L
Extra Slides
**CM-2 Ground Server Data Flow**

**Combustion Module Experiment Data Server (CMEDS)**

**Components:**

**Hardware:**
- Dual Processor Pentium IIs
- 512 MB memory
- 27 GB hard disk storage
- Hardware Disk Mirroring
- Uninterruptable Power Supply
- CD-ROM drive

**Software:**
- OS: Linux (Redhat distribution)
- Database: Informix
- Web Server: Apache
- Netbios: Samba
- GRC developed:
  - C, Java, PHP, HTML
  - Perl scripts, CGI, shell scripts

**Client Web Interface**
- Historic Data Query

**ODBC Interface**
- Lookout Client Near Real Time Data Query

**Samba Server**
- MS Windows File-based Access (GC, TIFF, FSD)

**Ground Command Computer**

**Data Stream Splitter**
- Uplink data
- Command
- AOS data
- Engr/Sci Data; File Downlink
- HRM Data; SpaceHab Data

**Database:**
- FTP LOS
  - Engr Data
- FTP LOS
  - Science Data

**DEPP Files**
- FTP LOS
  - DEPP Files

**DPP Images**
- FTP LOS
  - DPP Images

**SpaceHab Database**
- FTP LOS
  - SH Data

**Apache Web Server**
- Data Query Interface

**Glenn Research Center**

David Andrew Carek, P.E.

at Lewis Field
Author Biography

David Andrew Carek, P.E.

Mr. Carek is a senior research engineer in the Satellite Networks and Architectures Branch at the NASA Glenn Research Center. He is currently working on advanced communication system architecture concepts for the International Space Station. Prior to his current position, Mr. Carek managed flight software development for the Combustion Module-2 microgravity experiment. Mr. Carek has 15 years experience in computer systems integration and 7 years experience in flight software development and project management. He started his career at NASA performing structural analysis and developing computer-aided-engineering systems for advanced capabilities in structural analysis and design. Mr. Carek graduated magna cum laude in 1988 with a BSME from the University of Toledo, where he was selected as the “Outstanding Engineering Student of the Year” by the Ohio Society of Professional Engineers. Mr. Carek is a recipient of the “Silver Snoopy” award for outstanding efforts that contribute to the success of human space flight missions.