Autonomous Operations for Small, Low-Cost Missions: SME, EOR (STEDI), DATA, and Pluto Express

OUTLINE

• Introduction
• Challenges
• Solutions
• Technologies
  • Technology Gaps
• Visions for Future, Low-Cost Missions
  • Future Challenges
  • Envisioned Solutions
  • Tall Poles
• Summary
Introduction: Heritage from SME (Solar Mesospheric Explorer)

SME Mission Provides Heritage in Low-Cost Missions (1981-89)

- $17M for Total Mission
- $1.2M to Develop entire Mission Ops & Data System
- $500K / year Ops Budget
- 2.5 year development (on schedule)
- Operations Success
- Advances in Automation
  - Scheduling
  - Data Processing
  - Fault Monitoring
SME Heritage

• Advances in Interactive Ops
  • Near Realtime Science Optimization
  • Realtime health & status monitoring and response opportunity

• Lifecycle Continuity
  • One Mission Ops System for Prelaunch Test and Postlaunch Ops

• Student Controllers
  • Demonstrated quality of student work force
  • Demonstrated Educational Opportunities of University Ops Center
Current Experiences

- One Basic Set of Needs for End-to-End Mission Operations System (EEMOS) for all 3 missions (EOR-STEDI, DATA, Pluto Express)
- EOR (Educational Ozone Researcher)
  - Mission to Measure Total Ozone, including Low Altitudes, to Complement TOMS
  - Runner-Up Mission in STEDI competition
  - Low-Cost ($4.6M for spacecraft, payload, mission ops)
  - Faster (2-year development phase, for ‘97 launch)
Current Experiences

• DATA (Distribution and Automation Technology Advancement)
  • Shuttle Hitchhiker Mission to:
    • Measure Solar Irradiance & Dynamics
    • Demonstrate Ops System Technologies in Distributed Operations, Automation, and Autonomy
  • Sponsored by Code X INSTEP Program
    • Low Cost (<$500K)
    • Faster (2.5 year development)
Current Experiences

- **Pluto Express**
  - Two Sciencecraft to fly by Pluto and measure surface and atmosphere
    - Small and Low-Cost for Mission to Outer Planet
    - Strong requirement to reduce ops costs with long (~10 yr) cruise
  - One Basic Set of Needs for End-to-End Mission Operations System (“EEMOS”) for All Three Missions
Challenges

1. Interactive Science and Engineering:
   Enable mission scientists to quickly take advantage of “Opportunity Science / Targets of Opportunity” to increase scientific value of mission
   Enable mission engineers to check performance of spacecraft and instruments for prelaunch testing and post-launch diagnostics

2. Fault Detection / Correction:
   Enable on-board faults to be detected and responded to quickly to preserve mission health and to reduce any loss of observing time

3. Unpredictable Environment:
   Respond to on-board faults in an environment that is often unpredictable
Challenges

4. Updatable Operations Parameters:
   Enable operational constraints, rules, sequences, displays, and algorithms to be updated during the flight based on current performance

5. Migration / Evolution of Autonomy:
   Enable the level of automation and autonomy to be updated during the mission and to migrate from ground to space

6. Cooperative User-Machine Ops:
   Enable Mission Scientists and Engineers to work cooperatively with operations system located both in-space and on-the-ground
Challenges

7. Distributed Monitoring:
Enable students and other interested groups to monitor mission activities from their distributed locations

8. Remote Ops:
Operate a NASA Mission (or a Shuttle Payload) from a University

9. Reduce Costs:
... of Operating a space mission by reducing: number of ops personnel, number of shifts, and number of communication passes (and/or tracks). Enable these reductions through space and ground automation and autonomy
Challenges

10. Security:
    Provide a mission operations system that is secure against unauthorized users

11. Robust Systems:
    Provide robust and efficient systems both on-board and on the ground to support automation, autonomy, and user interactions

12. Distributed Ops Team:
    Enable a distributed team of ground operators, engineering analysts, and scientists to operate a space mission as a team
## Solutions & Technologies

<table>
<thead>
<tr>
<th>CHALLENGE</th>
<th>SOLUTION (Design Features)</th>
<th>TECHNOLOGY &amp; RATIONALE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. INTERACTIVE SCIENCE</td>
<td>Realtime Telemetry / Realtime Displays / Realtime Commanding</td>
<td>Internet links</td>
</tr>
<tr>
<td>2. FAULT DETECTION / CORRECTION</td>
<td>On-board &amp; Ground Software for Fault Detection, Isolation, and Response</td>
<td>Observation (SCL) &amp; Detection (SELMON) ’ Selection and Action (SCL) as Multi-Agent System</td>
</tr>
<tr>
<td>3. UNPREDICTABLE ENVIRONMENT</td>
<td>Rule-based, Intelligent Agent System</td>
<td>Observation (SCL) ’ Detection (SELMON) ’ Selection and Action (SCL) as Multi-Agent System (Sam’s chart)</td>
</tr>
<tr>
<td>4. UPDATABLE OPERATIONS</td>
<td>Updatable Database of limits, constraints, etc.</td>
<td>SCL</td>
</tr>
<tr>
<td>5. MIGRATION &amp; EVOLUTION OF AUTONOMY</td>
<td>Scripts added / applications added Database updates</td>
<td>SCL</td>
</tr>
<tr>
<td>6. COOPERATIVE USER-MACHINE OPS</td>
<td>Realtime Operating System with multiprocessing</td>
<td>RTEMS (Realtime Executive for Multiprocessor Systems)</td>
</tr>
</tbody>
</table>

Colorado Space Grant College
Hansen / Siewert
<table>
<thead>
<tr>
<th>CHALLENGE</th>
<th>SOLUTION (Design Features)</th>
<th>TECHNOLOGY &amp; RATIONALE</th>
</tr>
</thead>
<tbody>
<tr>
<td>7. DISTRIBUTED MONITORING</td>
<td>Internet access to data, WWW access to database</td>
<td>Internet, WWW</td>
</tr>
<tr>
<td>8. REMOTE OPS</td>
<td>Internet, Security techniques, Distributed Control Hierarchy</td>
<td>Internet, Kerberos, Firewall, SCL</td>
</tr>
<tr>
<td>9. REDUCE COSTS</td>
<td>On-Board and Ground Autonomy and Automation</td>
<td>SCL, SELMON, Rationalizer</td>
</tr>
<tr>
<td></td>
<td>Judicious use of OTS / COTS tools. Early and Evolutional prototypes</td>
<td>Spiral Development Model</td>
</tr>
<tr>
<td></td>
<td>Student development and ops teams</td>
<td></td>
</tr>
<tr>
<td>10. SECURITY</td>
<td>User Authentication, Transaction Authentication</td>
<td>Kerberos, Firewall</td>
</tr>
<tr>
<td>11. ROBUST SYSTEMS</td>
<td>Early and evolving prototypes</td>
<td>Spiral Development</td>
</tr>
<tr>
<td></td>
<td>Robust Operating System</td>
<td>RTEMS and Advanced Concepts</td>
</tr>
<tr>
<td>12. DISTRIBUTE OPS TEAM</td>
<td>Hierarchical Organization, SCL, Internet</td>
<td>SCL, Internet</td>
</tr>
</tbody>
</table>
Semi-Autonomous Operations Concept

- **Teleoperations**
  - Expensive continuous control and monitoring
  - Latency and bandwidth limitations complicate coordination and synchronization

- **Full Autonomy ("Zero Uplink")**
  - Local monitoring and control simplifies coordination and synchronization, but ...
  - Full autonomy difficult in “challenging environments”

- **Supervisory Control (Multi-Agent System)**
  - Distribution of monitoring and control between ground-segment operators, ground agent automation, and on-board agent automation
Operator Agent System

“Teleoperations - Continuous Monitoring and Control”
Autonomous Agent System

“Full Autonomy”

Perception → Action

Environment

Sensors

Effectors
Semi-Autonomous Multi-Agent System

Ground Segment

Perception → Action

Supervisory Control

On-board Autonomy

Teleoperations

Space Segment

Perception → Action

Ground Autonomy

Sensors

Effectors

Environment
Semi-Autonomous Modes of Operation

*Operator Agent* - Remote Sensors -> Operator Perception and Control -> Remote Effectors


*Monitoring Automation* - **Sensors -> *Automated Perception -> Operator Control -> Remote Effectors

*Supervisory Control* - **Sensors -> *Automated Perception -> Operator Supervision -> *Automated Action Selection -> *Automated Action -> **Effectors

*Autonomous Agent* - **Sensors -> *Automated Perception -> *Automated Action -> **Effectors

*Ground/Space Segment, ** Local/Remote
Detailed Multi-Agent System

Observation → Detection → Classification → Action Selection

Calibration & Filtering

Environment
Sensors
Effectors

Device Commands

Planning
Triggering & Scheduling

Reflex
Action Execution

Slower Goal-Oriented Response
Quickier Reactive Response

Colorado Space Grant College
Hansen / Siewert
Observation and Action System Support

Function

- planning
- “soft” real-time
- kernel-level “hard” real-time
- hardware control loops

Time Frame

- hours
- seconds
- milliseconds
- microseconds

Planning Agent
Soft Reflex Agent
Hard Reflex Agent

Planning Agent

Soft Reflex Agent

Hard Reflex Agent

Planning Agent

Soft Reflex Agent

Hard Reflex Agent

Planning Agent

Soft Reflex Agent

Hard Reflex Agent
DATA End-to-End Agent System Concept

Situated Agent
(High Bandwidth Full State)

Agent Surrogates
(Status)

Operator-Agent Interface
(Excepton Only)

Dist. Operations

Local Automation

Remote Automation

Supervisory Control

Manual Control

SENSORS

ACTUATORS

INTERSEGMNENT I/O SERVER

EMBEDDED Reflex Agent

GROUND Reflex Agent

GROUND Planning Agent
DATA Multi-Agent Implementation

• On-board Reactive Agent
  • SCL RTE performs ...
    • Action selection with rule-chaining and constraints
    • Actions from script and rule action database
  • Custom DeviceIO interface performs action execution

• Ground “Deliberative” Planning Agent
  • NASA JPL DATA-CHASER Planning System and Plan-IT II perform on-line
    • Heuristically guided iterative repair to schedule based on system model and constraints
    • Generates SCL rule activation and script scheduling commands
DATA Detailed Multi-Agent System

- **SCL RTE**
  - Database
  - Observation

- **SELMON**
  - Detection

- **SCL RTE**
  - "Rationalizer"
  - Rule-chaining

- **SCL RTE**
  - Rule-chaining
  - Action Selection

- **SCL RTE**
  - Reactive
  - Response

- **DCAPS & Plan-IT II**
  - Planning
  - Triggering & Scheduling

- **DeviceIO**
  - Device Commands (command translation)

- **DATA-CHASER Instruments**
  - Environment
    - Sensors
    - Effectors

- **DeviceIO**
  - Calibration & Filtering

Colorado Space Grant College
Hansen / Siewert
Implementation Gaps and Remedies

• SELMON written as tool in ground system
  • Updated to operate in on-board computer
  • Integrated with SCL
• Fault Detection, Classification, & Response not fully implemented on-board
  • SCL and SELMON integrated
  • Added Rationalizer and Action Selector to integrated on-board system
  • Adding mode-dependent performance characterization
• SCL has some bugs, undocumented features, and documentation deficiencies
  • Getting lots of help from ICS, Inc.
Implementation Gaps and Remedies

- Backup Comm (PPP modem)
- Scheduling Tool? Not integrated to Command & Control Tool
  - Integrated DCAPS/PLAN-IT-2 with SCL on Ground
- Security (Sun Firewall, Kerberos, and PAP)
- Test of Command & Control System needed before S/C done
  - Developed Spacecraft Simulator to enable early testing
- Training of Ops Team / Checkout of Ops procedures should not start with spacecraft
  - Testbed
Visions for Future, Low-Cost Mission Ops

- On-board computers, operating system, software same as ground computer
  - PC-like environment
  - Utilization of new distributed open COTS technologies
  - Flexible Real-Time Operating Systems kernels
    - Tailor to embedded system
    - Tailor to ground distributed computing environment

- PC-like spacecraft

- Constellation of Small Spacecraft to accomplish Virtual Missions
Visions for Future, Low-Cost Mission Ops

- Highly Interactive Scientist-in-the-Loop Ops
- Highly distributed experts on-call throughout network as part of Virtual Operations teams
- Colleges, K-12 Schools, and the General Public able to participate in missions by receiving live data, requesting sequences, and selective commanding of spacecraft
- Scientists work at “home institutions”