Lessons Learned in Control Center Technologies and Non-Technologies

Elaine R. Hansen
University of Colorado

for the Control Center Technology Conference
June 18, 19, 20, 1991
University of Houston — Clear Lake
Houston, Texas
SME Control Center and OASIS: Lessons Learned in Control Center Technologies and Non-Technologies

— Outline —

- The Solar Mesosphere Explorer (SME) Mission
- Features of the SME Control Center: Technical and Non-Technical
- Can these features be applied to other missions?
- OASIS: Software tools to support some common Control Center functions
SME Mission Operations Functions (simplified)
SME Control Center and OASIS:
Lessons Learned in Control Center Technologies and
Non-Technologies

— The Solar Mesosphere Explorer (SME) Mission —

Characteristics:
• To determine what causes ozone variations in our upper atmosphere
• A coordinated set of measurements
• Interactive science operations
• Realtime, quicklook, and in depth analysis
• Control Center located at University of Colorado - Boulder
SME Control Center and OASIS: Lessons Learned in Control Center Technologies and Non-Technologies

— The Solar Mesosphere Explorer (SME) Mission —

Results:

- Low cost: spacecraft, six science instruments, the entire ground data system, and one year of post launch operations for $17M
- Accomplished on schedule, within budget
- Strong personnel motivation
- All mission objectives met
- Control center performed well over the 7 1/2 year mission lifetime
SME Control Center and OASIS: Lessons Learned

— Features of the SME Control Center:
  Technical and Non-Technical —

1. University Based
2. Student Participation
3. Project Management
4. Integrated Design/Systems Design
5. Common Tools for Common Functions
6. Continuity over Project Lifecycle
7. Human Factors
1. University Based
   - Scientists and engineers able to work at their home institutions
   - Able to demonstrate advantages of "telescience" and "teleoperations"
   - Freedom to maintain and enhance system in response to changing mission, insights, and available technologies
2. Student Participation

- Major contributors to control center
  - 10 - 25 Undergraduate students per term
  - 2 - 4 Graduate students per term
- Variety of responsibilities
  - Controllers
  - Analysts — science and mission
  - Planners — science and mission
  - Teachers and tour guides
  - Programming assistants
  - Advanced development
- Providing perpetual motivation and ideas for enhancement
- Invaluable educational experience
Technical and Non-Technical Features of SME Control Center

3. Project Management — JPL

- Development teams motivated for on-orbit performance
  - Award fee
  - Science pay-off
  - Continuing operations responsibility

- Therefore, it was beneficial to
  - Help other teams
  - Maintain full and open communications between teams
  - Develop a reliable, usable, and maintainable operations system
  - Reveal and correct problems early

- Early involvement by control center designers

- Encouraged system level trades
  - Between science, instruments, spacecraft platform, control center, analysis system
  - To increase efficiency, reliability, capability and eliminate frills - with no effect on science objectives

- Simple interfaces between elements

- Project Management supportive of new operations approaches
— Technical and Non-Technical Features of SME Control Center —

4. Integrated Control Center Design
   • OK to develop new control center designs and tools
   • Top-down design approach
     — Based on science objectives and project requirements
     — For end-to-end (user-to-instrument) operations
     — For full lifecycle of operations support from early instrument tests through on-orbit operations
   • Functional elements defined from functional requirements
   • Functional interfaces to facilitate information exchange among elements
   • Functional elements arranged to minimize information interfaces
   • Processes and needs common to multiple elements identified
     — Common tools implemented
     — These tools duplicated for use in multiple elements
Technical and Non-Technical Features of SME Control Center

5. Common Tools for Common Functions

- Evident in continuing mission that even more control center functions are actually common and could be accomplished by common tools
- Evident that these common functions are not unique to SME Mission but are part of essentially all missions
General Task Functions Within SSF Ground System Nodes
Level of Commonality  
(by a grass roots evaluation)

<table>
<thead>
<tr>
<th>Task</th>
<th>Hardware</th>
<th>Software</th>
<th>Procedures</th>
<th>People</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command, Control &amp; Coordination</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>Storage, Handling &amp; Distribution</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>11</td>
</tr>
<tr>
<td>Scheduling</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>Planning</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>Analysis</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>System Maintenance &amp; Management</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>6</td>
</tr>
</tbody>
</table>

Commonality is defined as extent to which this task/function can be accomplished by a common set of hardware, software, or procedural tools or by people.

4 = Almost always  
3 = Generally  
2 = Sometimes  
1 = Hardly ever
6. Continuity over project lifecycle
   - Since common functions needed through lifecycle, a single operations system used throughout project lifecycle (prelaunch test, calibration, integration, and in-flight operations)
   - Benefits include:
     - Thorough and early system-level verification of the system hardware, software, procedures, facilities and personnel readiness
     - Early and continuing training
     - Control center bugs and enhancements determined and implemented early
     - Early test of the critical interfaces between major systems
     - Early and full access to capabilities of full operations system
     - Benefits in cost, schedule, reliability, and usability
Continuity Between Phases of SME Project Lifecycle is Maintained

**Instrument & Payload Integration & Test**

- **PLANNING** → **CONTROL CENTER**
- **ANALYSIS** ← **CONTROL CENTER**
- **BASIC COMMUNICATIONS SIMULATOR**
- **PROTOTYPE S/C COMMAND & DATA HANDLING SYSTEM**
- **INSTRUMENT(S)**

**Spacecraft & Satellite Integration & Test**

- **PLANNING** → **CONTROL CENTER**
- **ANALYSIS** ← **CONTROL CENTER**
- **BASIC COMMUNICATIONS SIMULATOR**
- **TEST OPERATIONS CONTROL CENTER (DUPLICATE CC)**
- **SATELLITE**

**Communication Compatibility Testing**

- **PLANNING** → **CONTROL CENTER**
- **ANALYSIS** ← **CONTROL CENTER**
- **NASA COMMUNICATIONS SYSTEM & SIMULATOR**
- **TEST OPERATIONS CONTROL CENTER**
- **SATELLITE**

**Flight Operations**

- **PLANNING** → **CONTROL CENTER**
- **ANALYSIS** ← **CONTROL CENTER**
- **NASA COMMUNICATIONS SYSTEM**
- **SATELLITE**
7. Human Factors

Control center elements composed of the following layers:

- People, user interfaces, facilities, procedures, software, and hardware
- People layer:
  - Defines people's roles and needs
  - Drives design of deeper layers
- SME users wanted to interact with the control center without going through intermediate programmers
  - Through interactive English-like languages and menus
  - Through user-specified, graphic displays
- Users preferred automation of tasks that are:
  - Predictable, routine, repeated, computational, critical, or potentially hazardous
  - But wanted ability to monitor most activities
The Control Center is designed in layers where the outer layers are used to drive the design of the deeper layers.
### SME Control Center and OASIS: Lessons Learned
— Can These Features be Applied to Other Missions? —

<table>
<thead>
<tr>
<th>Feature</th>
<th>Applicability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. University Based</td>
<td>• Yes, if appropriate&lt;br&gt;• Telescience/teleoperations/distributed operations approaches enable capabilities at user nodes</td>
</tr>
<tr>
<td>2. Student Participation</td>
<td>• Great if possible</td>
</tr>
<tr>
<td>3. Project Management</td>
<td>• Yes!</td>
</tr>
<tr>
<td>• Motivation</td>
<td></td>
</tr>
<tr>
<td>• Early involvement</td>
<td></td>
</tr>
<tr>
<td>• Systems trades</td>
<td></td>
</tr>
<tr>
<td>• Simplify interfaces</td>
<td></td>
</tr>
<tr>
<td>• Supportive of change</td>
<td></td>
</tr>
<tr>
<td>4. Integrated Design/Systems Design</td>
<td>• Yes! Seen as a major deficiency in current missions</td>
</tr>
</tbody>
</table>
| 5. Common Tools for Common Functions | • Yes, a major opportunity for lowering costs and increasing reliability  
• Biggest payoff at hardware and software layers  
• Little payoff in personnel and procedural layers  
• Results in interoperability between missions, within a mission, and throughout a mission's lifecycle |
| 6. Continuity over Project Lifecycle | • Yes, can see no technical or financial reason for not following |
| 7. Human Factors  
• Layered design to optimize work environment | • Yes, should be standard design technique |
OASIS Realtime Control and Monitoring Package "OASIS-RT"

- OASIS-RT allows scientists and engineers to control and monitor space instruments and subsystems throughout the entire project lifecycle
- OASIS-RT provides capabilities similar to those found in large spacecraft operations systems
- OASIS-RT is flexible and can be tailored to a particular application with no software changes
  - Procedures written by users in high level language
  - Spacecraft capabilities defined by database tables
  - User interface specified by database tables
- Ties with external diagnostic packages, analysis packages, etc.
- Coded in Ada
OASIS is Useful Throughout the Project Lifecycle

- During instrument test (connected directly to instrument)
- During systems integration and test (connected to remote test systems)
- During on-orbit operations (connected to remote project data systems)
How OASIS is Organized

User Interface

CSTOL

Display

Telemetry

Data Base

Command

Communication
OASIS — Planning and Scheduling Package "OASIS-PS"

- OASIS-PS allows scientists and engineers to plan and schedule experiments and subsystem activities throughout a program
- Case-based planning and scheduling system
  - Core systems independent of application
  - Designed to accept application-specific code
- Provides planning and scheduling in appropriate context for user
  - Application specific windows interact with scheduling components
  - Application specific database tables accessible by all components
- User defined "schedulers" attached to any scheduling timeline
  - Can have any type of scheduler tool, ie., expert system, heuristic, algorithmic, etc., working on a specific timeline in concert
  - Can have many different schedulers working at once
- Application driven communications protocol
- Coded in Ada
Summary

- Lessons learned from SME are indeed applicable to range of future missions
- Both technical and non-technical lessons are important
- Largest deficiency in today's systems seems to be a lack of an integrated systems perspective
- Software toolsets are available today to support some of these common control center functions