CUBESAT
California Polytechnic State University

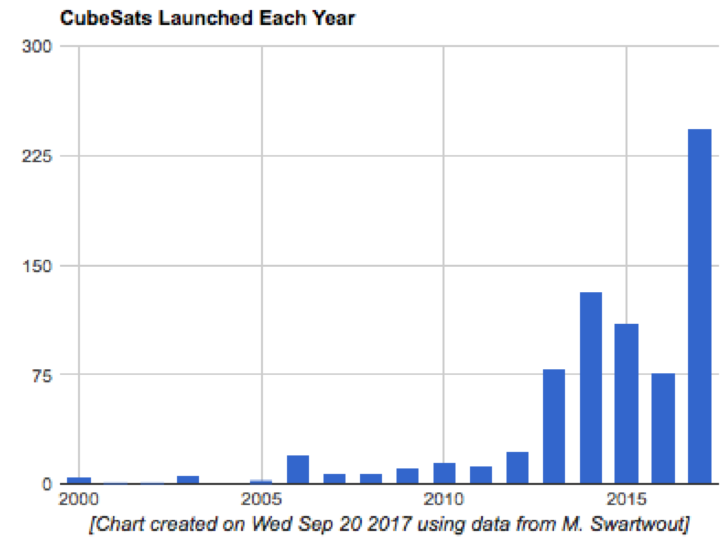
CubeSat/Small Satellite Lessons Learned

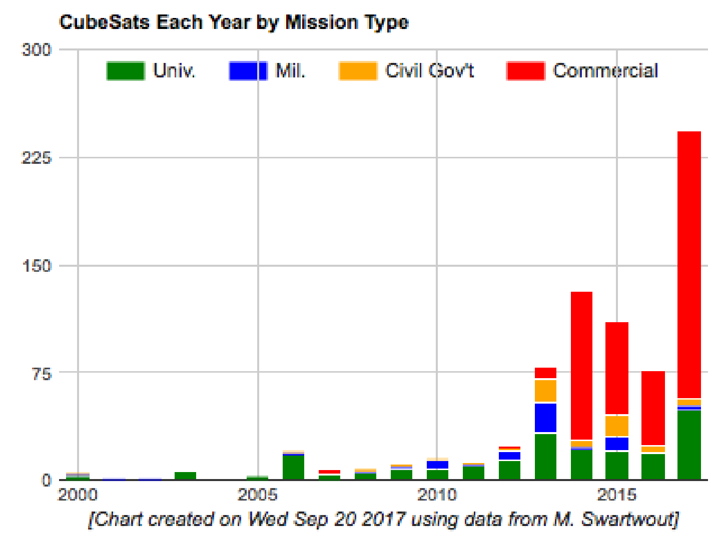
Ryan Nugent
California Polytechnic State University, San Luis Obispo

5th UNISEC Global Meeting, Rome, Italy
December 3rd, 2017

CubeSat Community Current State

- Thousands of developers worldwide
  - Across academia, industry, and government agencies
- 750+ CubeSats have been launched
- No longer only an academic training tool or industry testbed
- Constant new entrants
  - New countries, new universities, new companies





New Entrants Common Issues

- Common failures include deployable failures, power system failures, comm system failures
- First time university developers have low mission success rates
  - Most CubeSats are not able to be contacted
    - Difficult to determine cause of anomaly
- New entrants tend to make the same 'first-timer' mistakes
  - Most are easily avoidable
- How do we increase new entrant mission success?

Lessons Learned Discussion Main Topics

- Set Appropriate Scope for the Mission
- Establishing Program Structure
- Schedule
- Risk Management
- Design, Production, and Assembly
- Assembly
- Test, Test, Test
- Operations

Group Discussion Summary

- Establish minimum baseline mission with modest success criteria
  - Stick to and defend these minimum requirements and goals
    - Do **NOT** allow requirements to be added after requirements have been decided
- Build an experienced team
- Rigorous documentation and reviews are important
  - Helps to maintain continuity of knowledge
- Reviews are very important
  - Independent reviewers are very helpful

Group Discussion Summary

- Maintain Schedule and Margin
  - Most on orbit anomalies attributed to lack of testing on the ground
- A good risk management process is very important for CubeSats
- Design for simplicity and robustness
- Test Early and Often
  - Performing fully integrated testing early will catch the most anomalies, greatly increases mission success

Group Discussion Notes

Set an Appropriate Scope for the Mission

- Establish a minimum baseline mission with modest success criteria
  - May have de-scope plan in place should one be needed
  - Stick to and defend these minimum requirements and goals
    - Do **NOT** allow requirements to be added after requirements have been decided
- Develop simplest spacecraft to fulfill mission
  - Cal Poly/JPL IPEX mission, started as a 3U with 3-axis ADC, ended as a 1U with no ADC
    - Met all mission requirements
- Stick with your expertise, don't do new science

Establishing Program Structure

- Build an experienced team
  - Teams with more experience tend to have higher success rates
    - Mentors from industry help apply best practices and lessons learned to academic programs
    - Focus on the team's strengths and interests
    - Systems engineering is very important, must have during all phases of the project from conceptual design through operations
- Rigorous Documentation
  - University teams have high turnover rates, helps maintain continuity of knowledge
- Reviews
  - Necessary evil, don't need to be formal but do need to be rigorous
    - Independent reviewers are very helpful
- Have a small core team that communicates regularly
- At the university level, plan for turnover of students (they will graduate eventually)
  - Senior students should always train a younger student on their tasks

Schedule

- Assembly, Integration, and Test should be a large portion of the schedule
  - Maintain this portion of the schedule, as this is where anomalies are found
- Launches don't wait for CubeSats, be ready for the schedule crunch that will happen as delivery gets closer
  - Puts extreme pressure on the latter half of the schedule, usually on AI&T
  - Have margin and contingency plans to maintain mission success
  - Work with Launch Integrator to find more schedule in case necessary
- Stick to your schedule, create milestones and stick to them

Risk Management

- A good risk management process is very important for CubeSats
  - Do a risk assessment at the beginning
    - What is new? What is single point of failure?
- Purchase multiple sets of hardware
  - Use for 'drop in' replacements in case of failure, minimize schedule risk
- Software is always risk
  - Early functional testing is necessary
- Risk to cost ratio
  - When choosing analyses or tests to perform, focus on easiest to solve and work up from there

Design, Production, Assembly

- Design for simplicity and robustness
  - Minimize deployables and keep them simple
  - Design for the worst case environment
- Employ fail-safes built into the satellite electronics
  - Watch-dog timers, planned resets
  - Define what your safe mode, make sure your satellite can recover from safe mode
- Design for disassembly and re-work
  - Many issues are not discovered until the satellite is fully assembled
- Overdesign and overbuild for risk reduction
  - Manufacture or purchase extra parts, testing anomalies or mishandling of equipment will happen
  - Don't design to the specifications in COTS components datasheets, apply values to de-rate them as appropriate, also test them to see how they behave
- Always have an omni directional antenna, at least as a back up

Design, Production, Assembly

- Perform inspection of all parts when they are received
- Clearly define tolerances
- Use 2 back out prevention methods for all fasteners
- Always check
  - Electronics that CubeSat become obsolete quickly
  - Don’t underestimate lead time for any component no matter how simple

Test, Test, Test!

- Subsystem testing important, integrated system testing is the most important
  - Most on orbit failures attributed to lack of integrated system testing
  - No matter how much time you have scheduled for testing, it won't be enough
  - Cal Poly develops a 'flatsat,' engineering test unit, and flight unit for most missions
- Performing full end-to-end system testing is important to perform as early as possible
  - Examples include: Command execution testing, Day In the Life tests, End-to-end Comms testing, Full Power system charge cycle
- Thermal Vacuum Testing best simulates space environment
  - Resource intensive, if T-vac not available, perform testing at temperature extremes in ambient conditions

Operations

- Don’t underestimate the the difficulty of tracking and commanding a CubeSat
- Ground segment should be developed in parallel or before the CubeSat
- Analyze trends of your satellite
  - Battery degradation, temperatures, etc.
- Practice operations on your own CubeSat and other CubeSats
  - Insert errors to see how operators respond