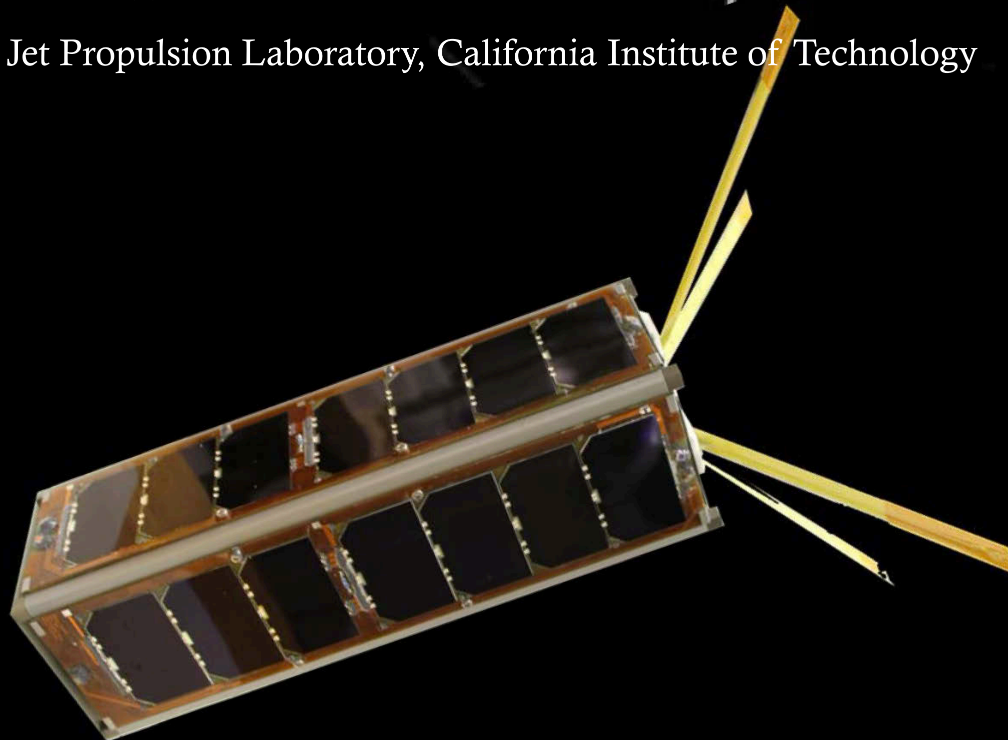
CubeSat Challenge Team

Using MBSE for Operational Analysis

In Affiliation with the Jet Propulsion Laboratory, California Institute of Technology



CubeSat Challenge Team

- JPL
  - Louise Anderson
  - Christopher Delp
  - Elyse Fosse
  - Bjorn Cole
  - Leo Cheng
- Michigan
  - Sara Spangelo
  - James Cutler
- agi — Analysis software for land, sea, air, & space
  - David Kaslow
- No Magic
- Phoenix Integration
  - Grant Soremekun
- icx
  - Manas Bajaj
  - Rose Yntema

Agenda

- MBSE Initiative
- Motivation
- Radio Explorer Mission (RAX) using SysML
- Overview of Analysis Demonstrations
- Analysis Demonstrations
  - Operations Prediction of State
    - Phoenix Integration - ModelCenter
  - Spacecraft Control (Flight-Ground)
    - Cameo Simulation Toolkit
  - A-causal Analysis & Requirements
    - Paramagic
  - Document Generation and Editing
- Results
- Future Plans

INCOSE Space Systems Challenge Team

- Produce a Space Systems example of MBSE
  - Real-world
  - Sharable
    - INCOSE is an International Organization
- Broad Team
  - NASA, CSA
  - Industry
  - Academia (MIT, GIT)

INCOSE Space Systems Challenge Team

- 5 years of MBSE Investigations and Demos
  - FireSat Model
    - Industry and Academic participation
  - Space Systems Library
    - Parametric Library based on SMAD (Wertz, Larson)
  - CubeSat
    - Framework
    - Integrated Tools

Motivation

Driving Idea: Systems Engineering Specifications

- Systems Engineers in the Space Domain Produce Information
  - Functions
    - Describing the Problem
    - Trades
    - System Functional and Behavioral Design
    - Specifying System Components and Integrations
    - VnV
    - Deployment and Fielding
    - Operational Support
  - Products
    - Analysis (Simulation, Tests, etc)
    - Reports on Analysis
    - Plans
    - Design Descriptions
    - Interface Descriptions
    - Requirements
- *Relationships are primarily inter-disciplinary*

Challenge: Communication and Consistency

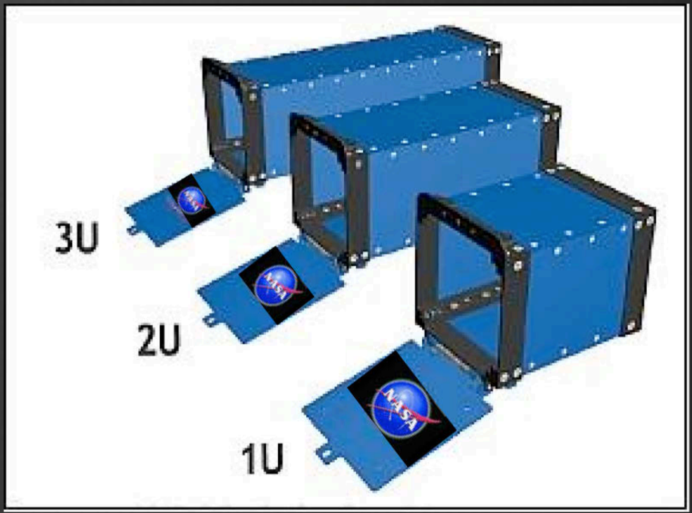
- Challenges
  - Communicating the system in a world of models
    - How do you extract all the rich detail from these simulations into System Specification?
    - DOORS? Documents/Slides/Spreadsheets?
    - How do you assert mutual consistency between models?
    - Meetings? Emails?
- Need an equally rich mechanism for expressing the system design
  - Human readable
  - Machine readable

Role of Languages in MBSE Enterprise

- Capture and Express Information
- SysML (Systems Modeling Language)
  - Unifying the Qualitative and Quantitative
  - Descriptive Information
  - *Designed* to serve as *System Specification*
  - "Machineable"
    - Compatible with a variety of execution and analysis strategies
- Authoritative Source of information

CubeSats?

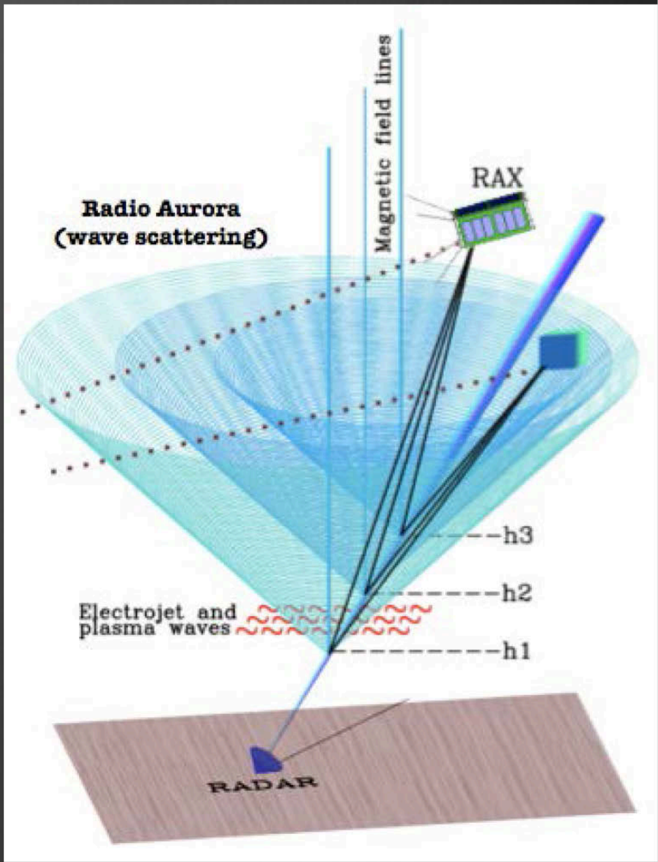
- NanoSatellite (1-10kg)
  - Used for Space Research
- 1U = 10 cm^3
- Ultra Low Cost Missions
  - University/Company Training
  - COTS Hardware
- First CubeSat Launched in 2003
  - Over 75+ CubeSats in Operations



ISIS. "CubeSat Concept - Satellite Missions." *CubeSat Concept - Satellite Missions*. N.p., n.d. Web. 13 Jan. 2013. <https://directory.eoportal.org/web/eoportal/satellite-missions/c-missions/CubeSat-concept>

Radio Aurora Explorer (RAX)

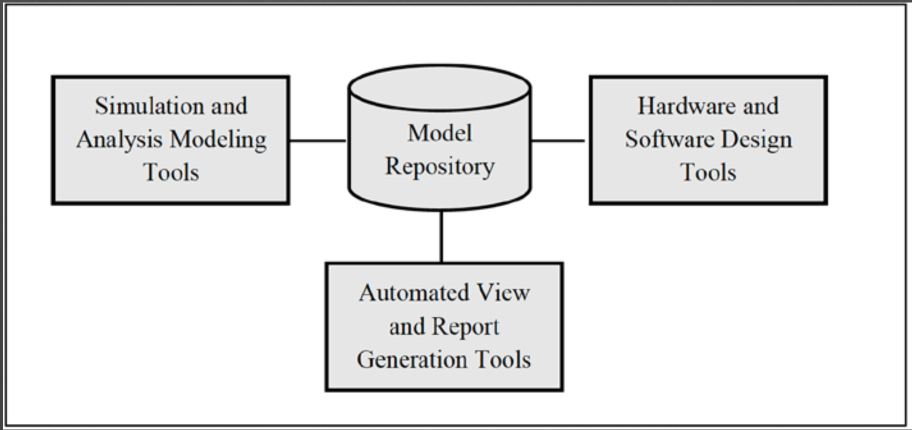
- Michigan Exploration Lab and SRI International CubeSat mission
- Study formation of magnetic field aligned plasma irregularities in the lower polar ionosphere
  - Disturbs ground-space communication and navigation
- Radar signal is transmitted by Incoherent Scatter Radar site in Poker Flat, Alaska and received by RAX for analysis
  - Science data processed on-board, compressed, and is downloaded to a globally distributed network and is commanded from the control center in Ann Arbor, Michigan



Demonstration Overview

- System Model Description
  - CubeSat Framework
  - RAX Implementation
- Power Prediction Analysis
  - Power Loads Analysis Driven by operational scenario
- Spacecraft Behavior Prediction Analysis
  - Spacecraft State Analysis Driven by operational scenario
- Communication Design and Requirements Analysis
  - Determine design criteria and constraints based on ground system architecture and design parameters
- Document Generation and Reporting
  - Document and reports of model and analysis

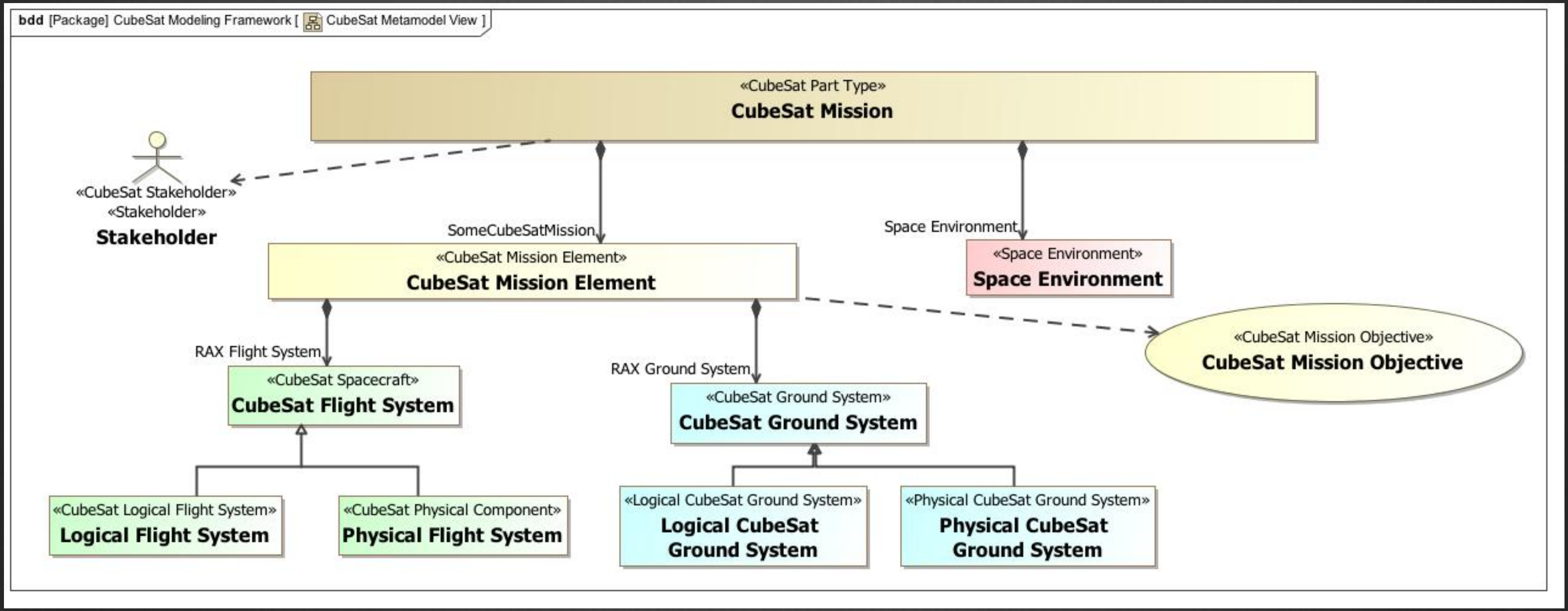
Value of Integrated MBEE



- Authoritative Source
- Standard Based Communication and Description
- Relatively Low Cost

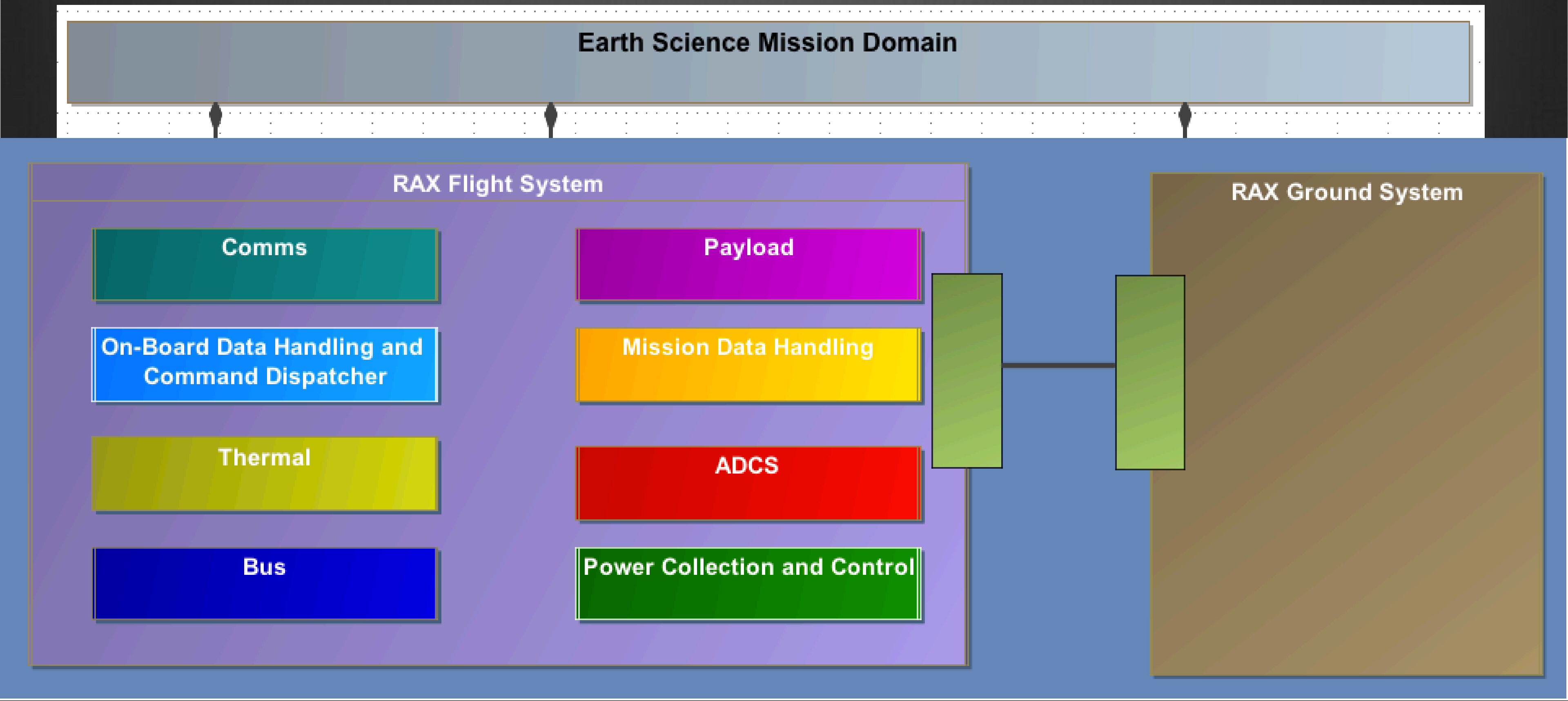
- Analyze with COTS Tools
  - MagicDraw (SysML), Systems Tool Kit (STK), Matlab
    - Phoenix ModelCenter (PHX) acts as "glue"
  - Cameo Simulation Toolkit, system behavior execution
  - Intercax- Paramagic and mathematica

CubeSat Framework



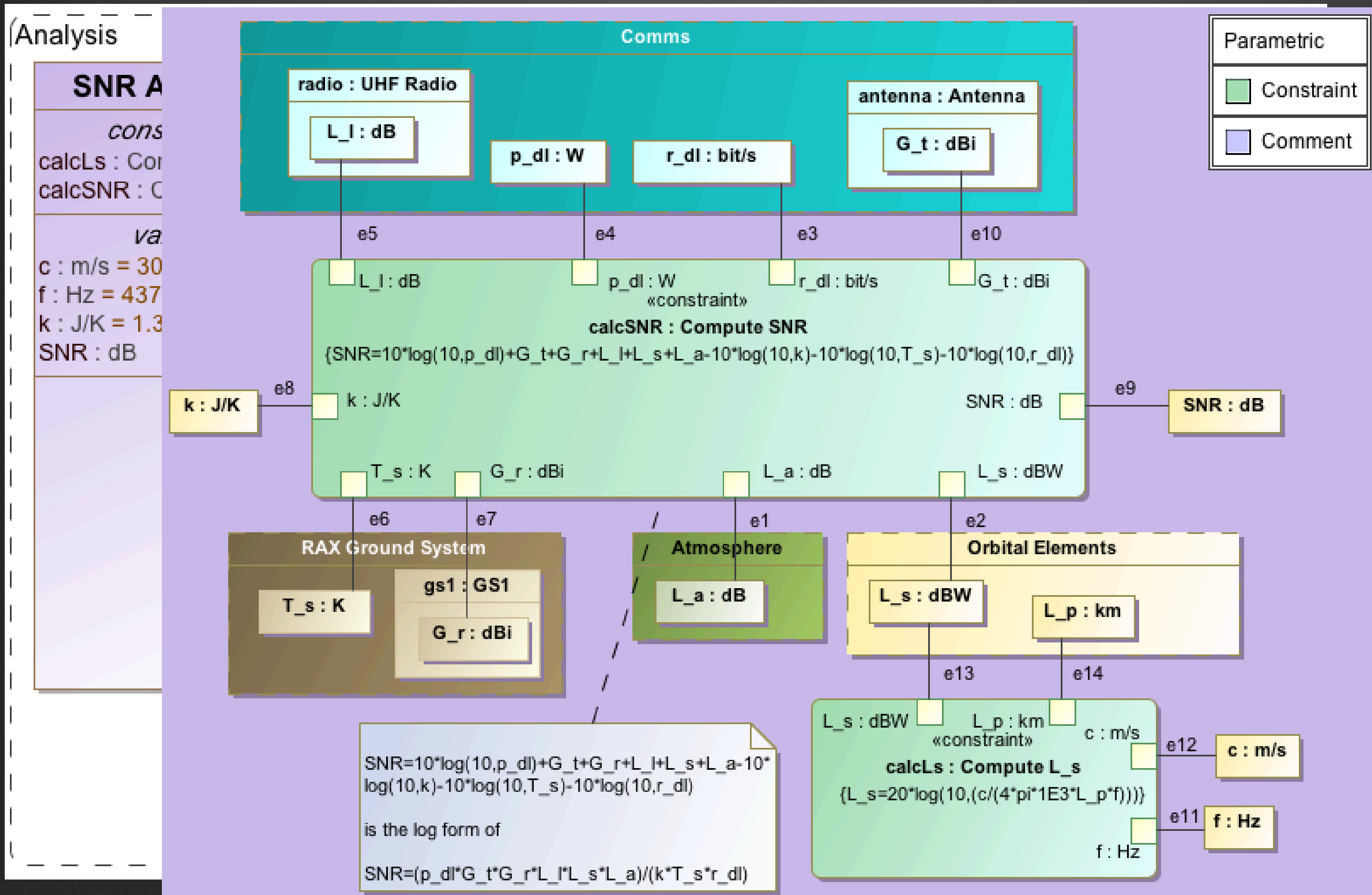
- Mission Adaptable
- Each piece of CubeSat Mission modeled
  - Environment, Flight, Ground

Modeling RAX



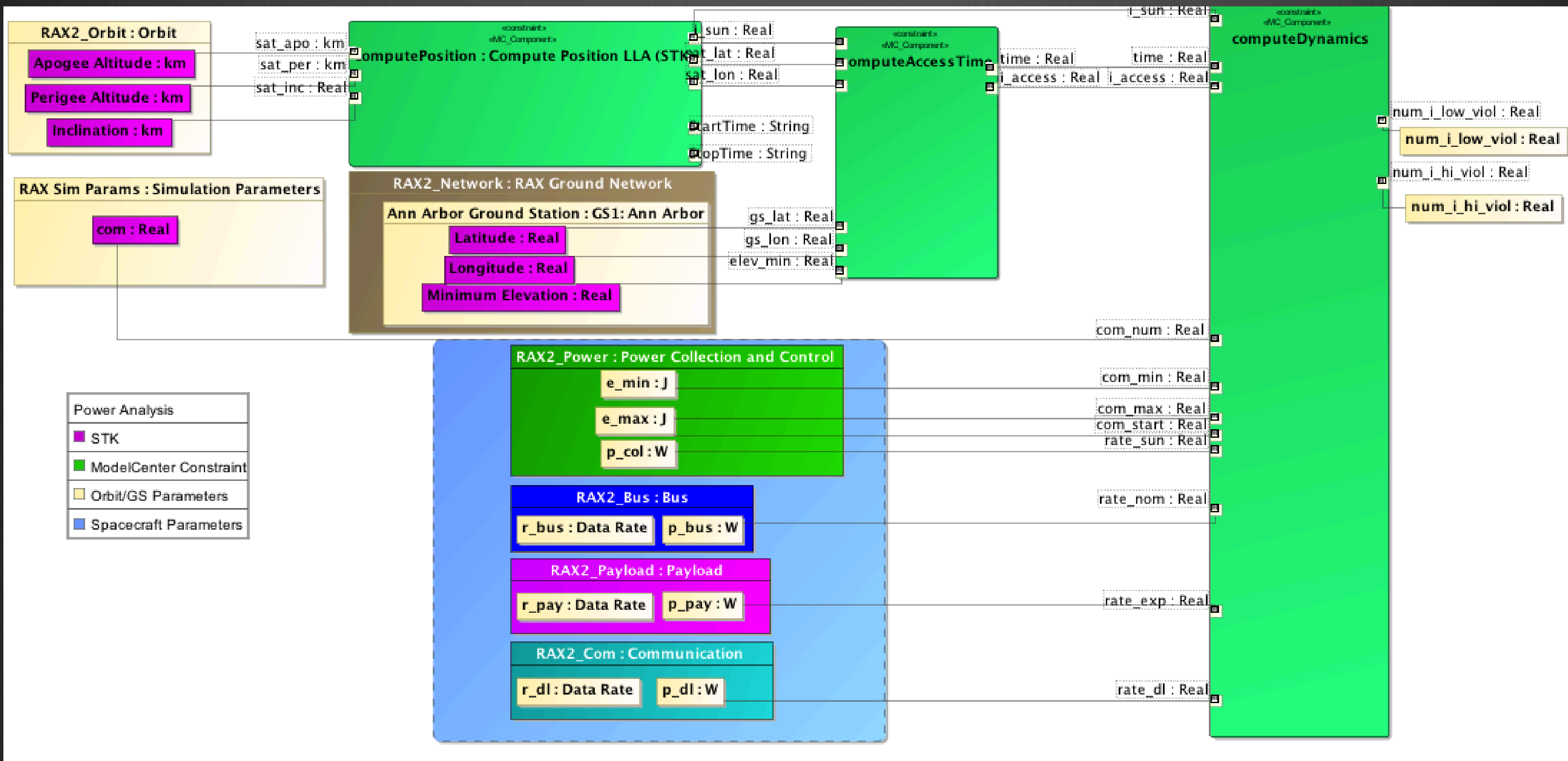
Communication Subsystem

- Communication Subsystem – Signal to Noise Ratio Analysis
  - SysML Parametrics

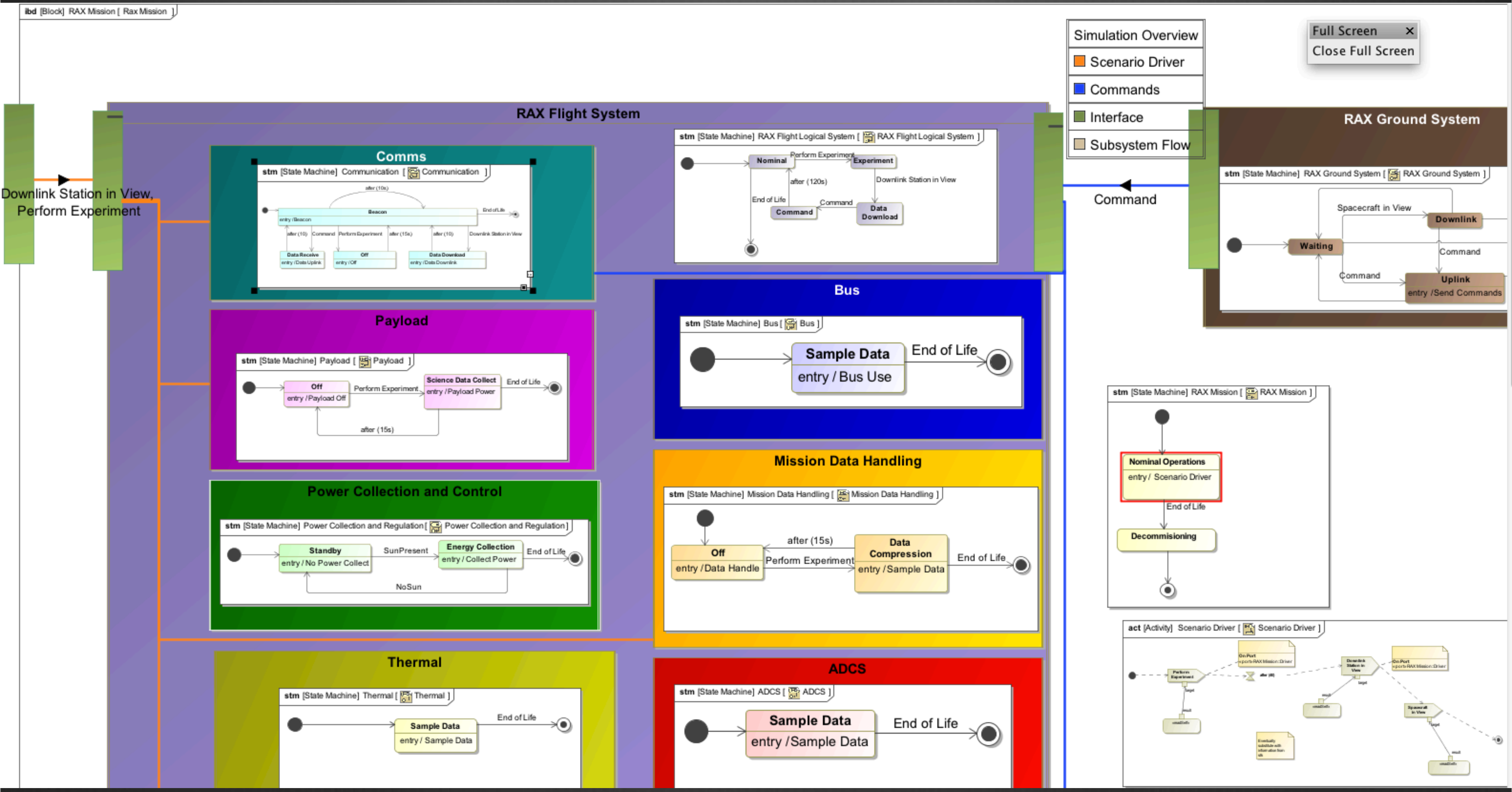


Power Analysis

- Power Subsystem – Power Analysis
  - SysML Parametric



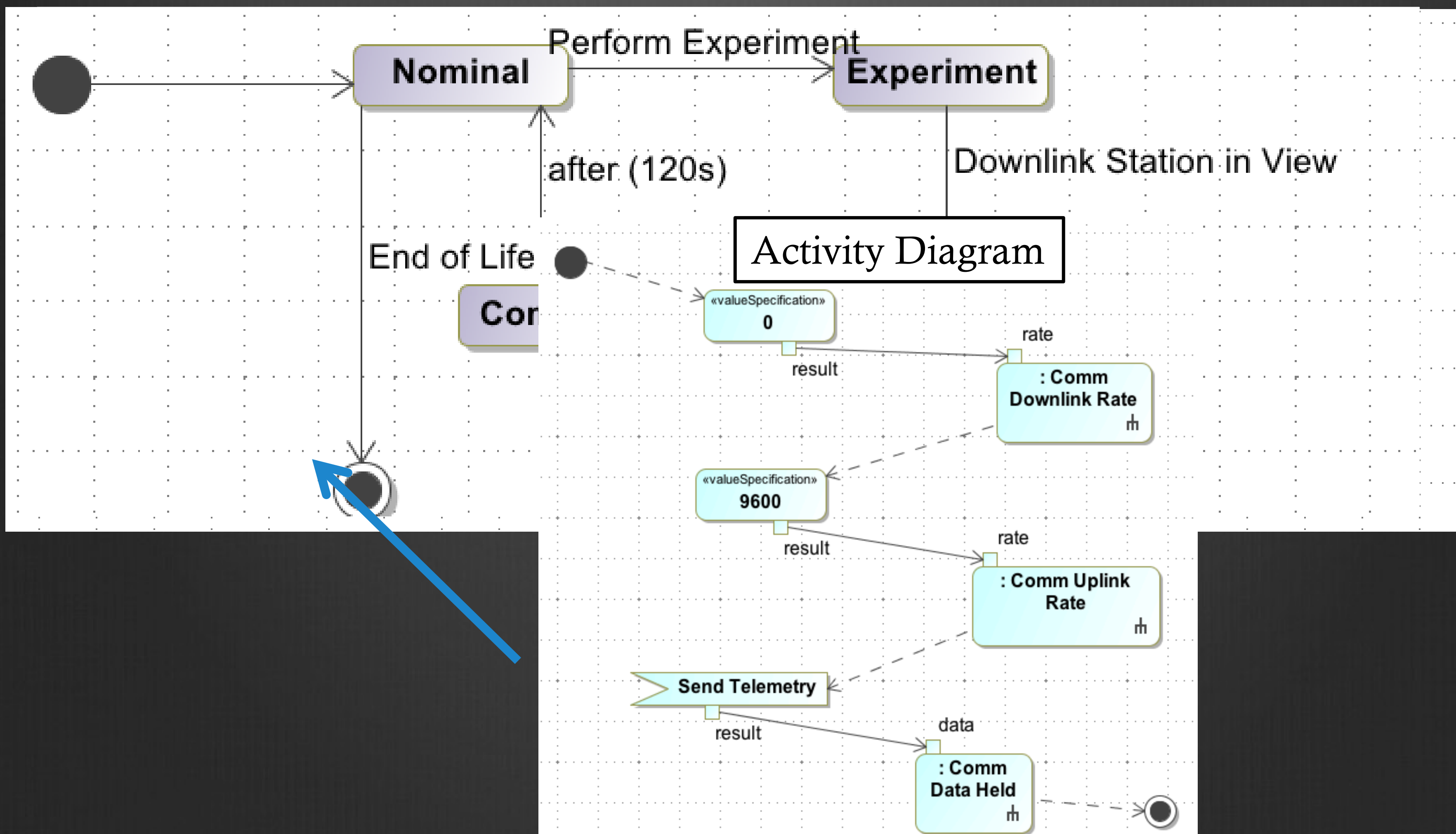
System Components & Behaviors



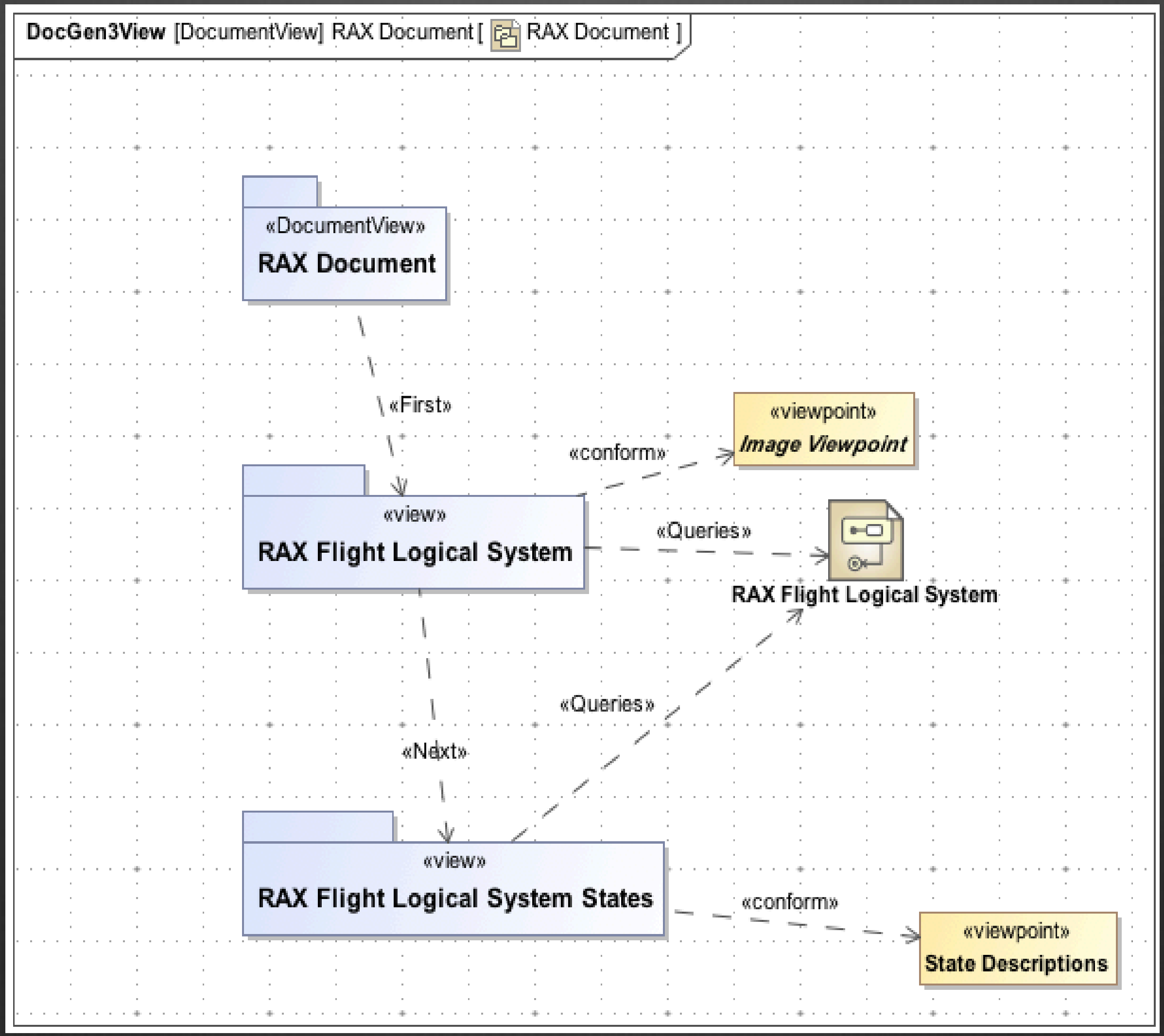
RAX Flight System Behavior

State Machine

Flight System States



RAX Report Generation



Demo Videos

Results & Successes

- Effectively described different views of RAX System
- Analyzed the RAX Model according to common practices in Space Systems Engineering
- Analysis completed with COTS Tools
  - Integrated around standard SysML Language
- Demonstrated capability to generate Documents
- "Develop With What you Fly With" End to End Integration and Analysis

Summary of Issues and Challenges

- Gaps in what can be effectively integrated with different analysis tools
  - Show independent analysis
    - No way to orchestrate inter-operation
    - Example: Cameo Simulation Tookit integration with Phoenix ModelCenter
      - Behavioral and structural analysis working together
- Both SysML and MBSE analysis tools are limited in temporal semantics
  - Time is key factor for space systems engineering

Acknowledgements

QUESTIONS?

Backup

- Successes and Challenge for each demo

Power Scenario Demo Overview

Motivation

- "Bringing the model to life", executing model
- Replaces "hacked" integrated software (e.g. manual/ complex code)

Integrating multiple software tools

- MagicDraw (SysML), Systems Tool Kit (STK), Matlab
- Phoenix ModelCenter (PHX) acts as "glue"

What does this enable?

- "Batch" execution of scenarios (i.e. full time history at once)
- Evaluation if requirements are satisfied/objective
- Test/compare scheduling algorithms (heuristic, optimized, etc.)
- Automatically re-run different scenarios (e.g. vary orbit, network)
- Parametric studies: Sensitivity to vehicle/ network parameters

Power Scenario: Lessons Learned

Useful things we "figured out" (with vendor support):

- Extracting time-dependent parameters (e.g. position in STK)
- Passing vectors between simulators was equally useful in PHX

Things to keep in mind for future modeling:

- Ensure you have required licenses! (may require vendor support)
- Parametric diagrams must inherit inputs/ outputs of PHX models
- Exploit existing code/ scenarios as much as possible
- Maintain modularity so can re-configure code for different applications

Document Generation

- Models are great, still need to support reviews and presentations
- Generate document artifacts from the model
  - Leverage ISO 42010 (with some extensions)
- Domain Specific Experts and Reviewers should *NOT* have to go back to the model to do their job
- Need a way to present the model-based document artifacts to others without requiring others to understand the model.

Challenge Team History

- 2007 – First Challenge Team was Founded
- 2007-2010
  - SysML Model of FireSat (SMAD Textbook)
    - SysML Suitability for modeling space missions
- 2011 – CubeSat Initiative Began
  - CubeSat Modeling Framework
  - Foundation to model/design many current and future CubeSat missions
- 2012
  - Applying SysML Framework to Operational Mission

Timeline of Activity

- Y1: MIT/GaTech Student FireSat Example
- Y2-4: SysML model of FireSat
  - Space Analysis Library using SMAD (Space Mission Analysis and Design textbook, Wertz and Larson)
  - Basic Model of FireSat
  - Solar Panel Trade
  - Satellite Toolkit Integration
- Y5->: CubeSat: An Architecture Framework and Method for Space Systems MBSE

FireSat MIT/GaTech Collaboration

- Build an integrated model of FireSat
  - SubSystems in Matlab, STK, Excel
  - Integrated with Phoenix Model Center
  - Student Teams Mentored by Industry Experts from INCOSE SSWG
- Successes
  - executable trade model for FireSat
- Challenges
  - Difficult to build
  - SubSystem models were difficult to integrate
  - No architecture of the model integration or key parameters
  - Difficult to Audit for completeness correctness

FireSat SysML Model

- Build SysML model of FireSat
  - Learn SysML
  - Describe FireSat using SysML
  - Compare Model Description against typical document representation
- Successes
  - Models of descriptions from book
  - Model views corresponding to documents
- Challenges
  - Technique of modeling and applying the methodology
  - Table representations
  - Model Analysis
  - Document Production

SysML Space Analysis Library

- Build SysML Space Analysis Library
  - Build Library of analysis from SMAD
  - Build approach to VnV for Library
- Successes
  - Libraries for many analysis types
  - Useful testing approach
- Challenges
  - Deep subject – much could not be captured
  - Executability (significantly improved since)
  - Units and Dimensions (significantly improved since)
  - Presentation of equations

FireSat Solar Panel Trade

- Use Library to replicate Solar Panel Sizing Trade
  - FireSat Model and Library-> executable trade
- Successes
  - Successfully built executable trade
  - Hard-linked to requirements
  - Powerful view of driving systems properties
- Challenges
  - Executability (improving since)
    - Debugging
    - Scaling

FireSat Integrated Modeling

- Integrate FireSat SysML Model with Satellite ToolKit
  - Exchange Orbit Scenario properties
- Successes
  - Basic Exchange of Parameters
  - Direct comparison of MBSE in SysML and STK
  - Explicit link between models and requirements
- Challenges
  - Integration
    - Complicated
    - Difficult to Scale

CubeSat: Framework and Method

- Build a Modeling Framework and Method for CubeSats
  - CubeSat Domain-Specific Terms
  - SE Framework for Modeling CubeSat Missions, Spacecraft, and Ground Systems
  - Example Application using RAX Mission
- Successes
  - First version of Framework
  - Early version of multiple executable demos
- Challenges
  - Resources
  - Executability
  - Integration

Consensus of Team

- Modeling with SysML
  - Everything was hard at first
  - Methodology is critical to a model that hangs together
  - SysML simplified construction of basic things like functions and properties
  - SysML tastes like early CAD apps
  - Libraries of model analysis were effective in making solar panel trade
  - Integration with STK
- Document Comparison
  - Model unified properties between views
  - Simplified understanding of the System
  - The common SysML language improved communication between teams and simplified collaboration
  - Automated reports allowed for more time to focus on engineering