



# **MBSE Experiences at LM-SSC**

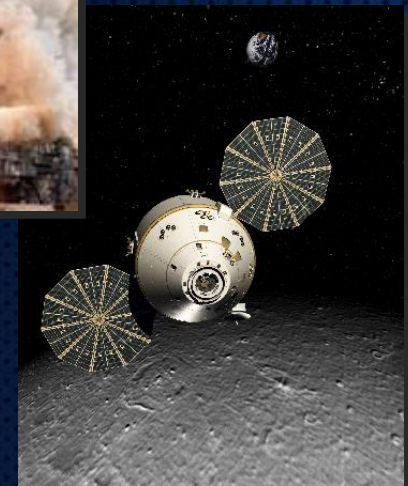
**for**

## **INCOSE MBSE Workshop IW2015**

Chris Schreiber  
Systems Engineering

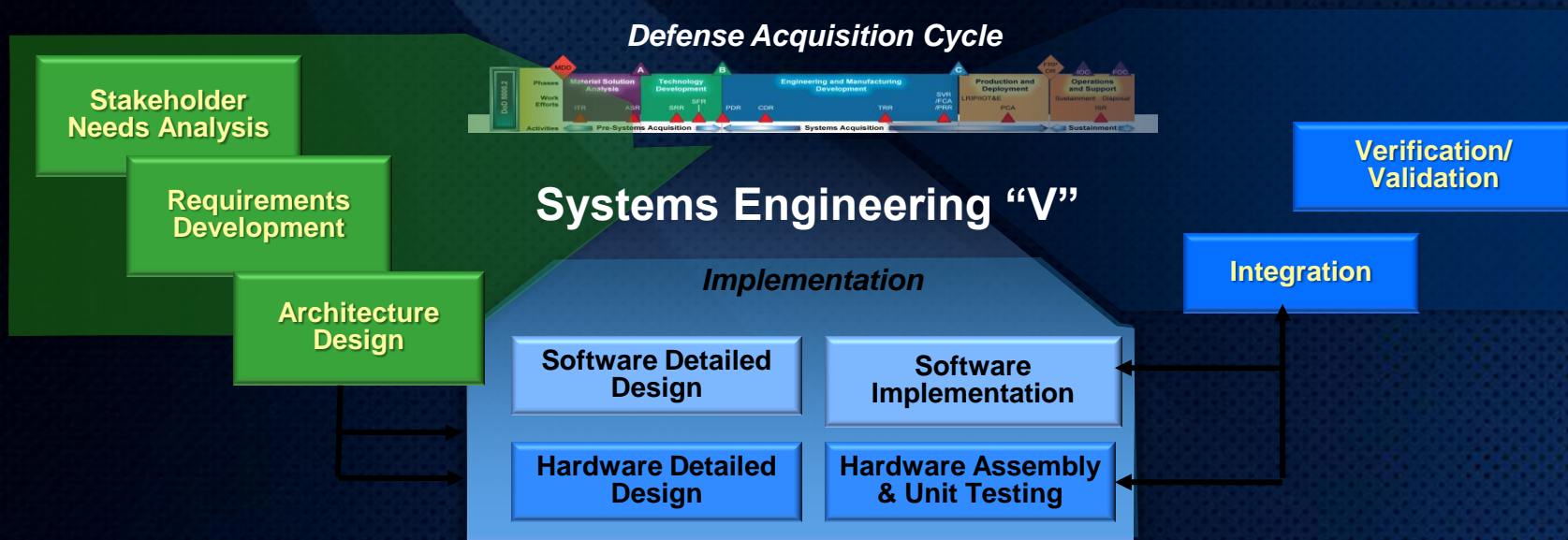
# Trends in Modern Systems

- Increased system breadth and complexity
- Emergence of Systems of Systems
- Increased attention to cyber security and system resilience
- Blurred line between hardware and software
- Customers focused on Affordability, Flexibility and Adaptability



***System Complexity Increasing with Budgets Decreasing***

# Systems Engineering Process



Plus . . .

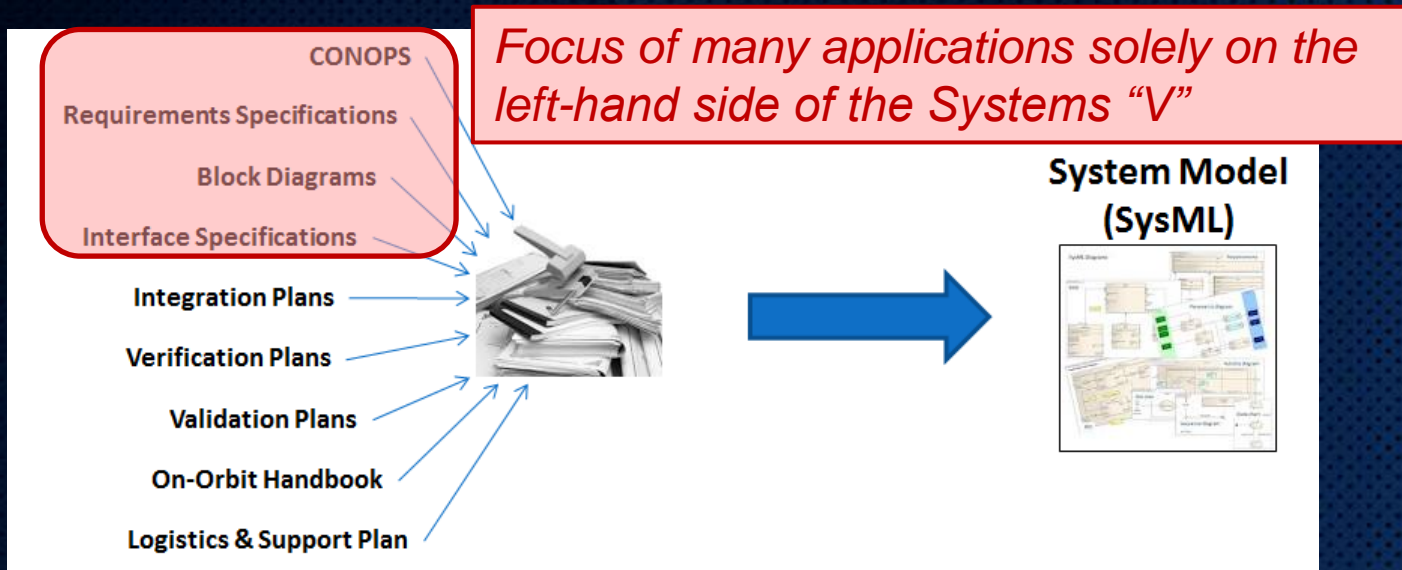
- Operate System
- Support / Maintain System
- Dispose System
- Risk Management
- Peer Review
- Decision Analysis
- Failure Review



# What is Model-Based Systems Engineering (MBSE)?

- “Model-based systems engineering (MBSE) is the **formalized application of modeling** to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases.”

– INCOSE SE Vision 2020 (INCOSE-TP-2004-004-02), Sept 2007

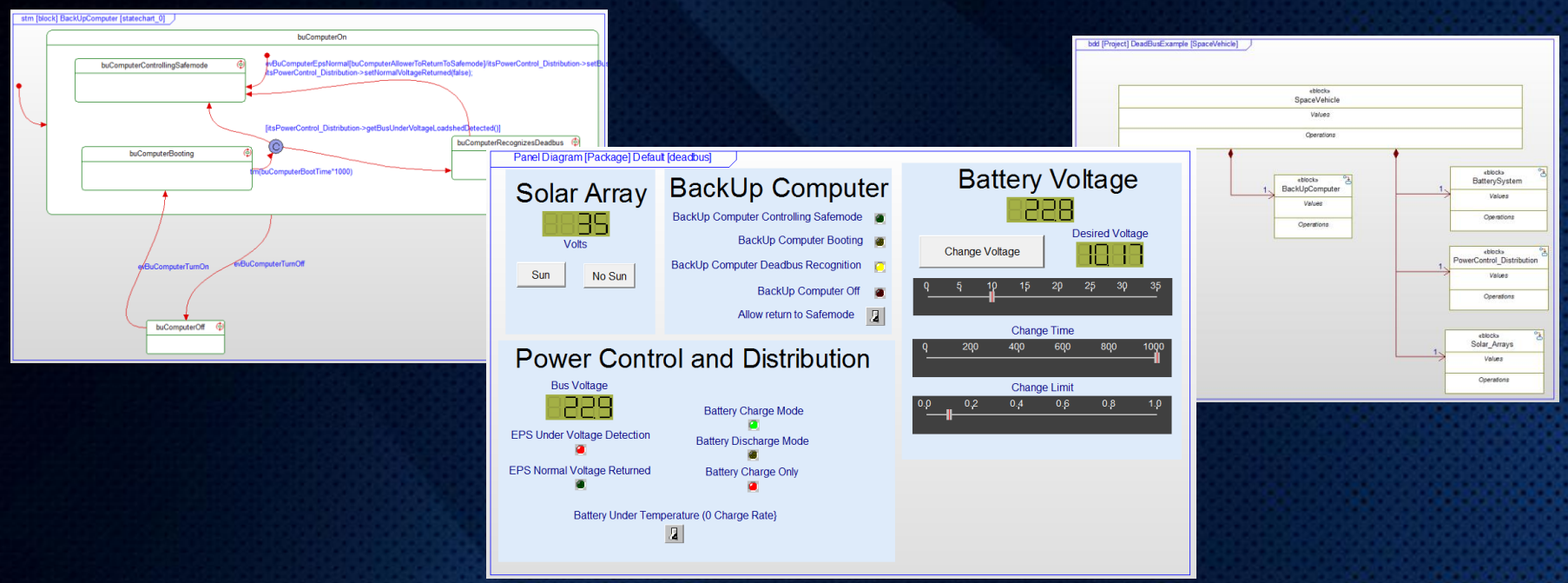


**MBSE is SE! . . . Using a model**



# Stakeholder Needs & Requirements

- State Machine modeling allows analysis on separate components that must act together into an executable dynamic scenario
- Provides “stop-motion” analysis so that commands, telemetry, and sensor states can be simultaneously examined at moments in time

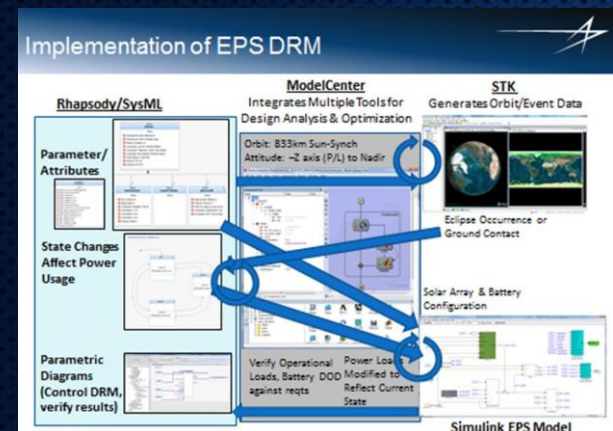
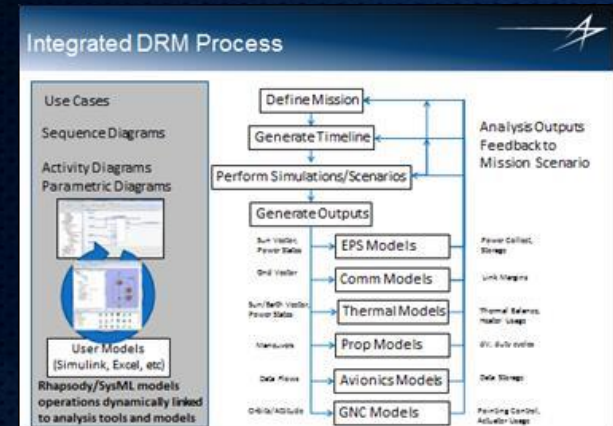


**Graphical Modeling Approaches Improve Quality & Enhance Communication**



# Design & Analysis

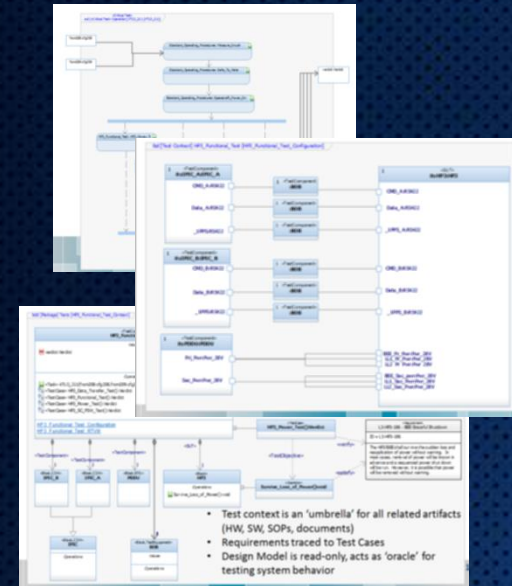
- Increased Fidelity and Consistency of information for analysis
  - System model key to integrated data from different aspects of design into a system level analysis (STK, Simulink, Rhapsody)
- Ability to explore different system designs against mission profiles
  - Repeatable analyses to evaluate alternatives against TPM's
- Foundational to use of trade space optimization tools
  - Analysis of Alternatives and Design of Experiments possible extending tools used to integrate the analyses (Phoenix ModelCenter Optimization Pak has DoE capability)



# Verification & Validation



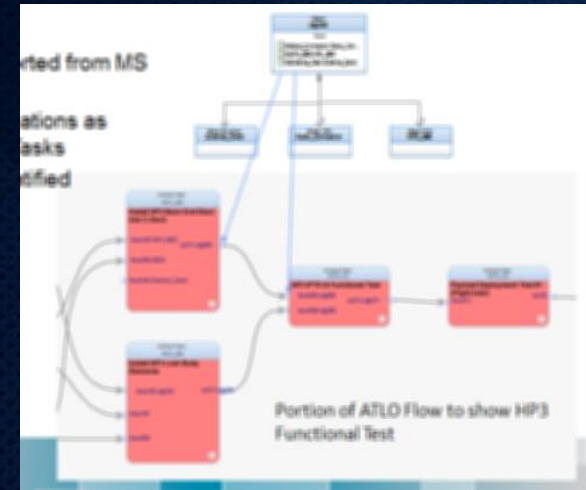
- Experience implementing a modified version of UTP
  - Value as a structured method for defining test and as input to test planning process
  - Especially useful for verification at system level (maybe less for component test)
- Platform for Test Planning Conversations
  - Cross-function discussions around test events
  - Definition for test support equipment
  - Integrate test with integration defining AI&T flow



# Integration



- System Structure based version for assembly sequence
- Model-based traceability for behavior and structure for integration and test (system configuration)
- Basis for linkage between detail design disciplines (mechanical, electrical, software, etc.) for manufacture/assembly and systems/software

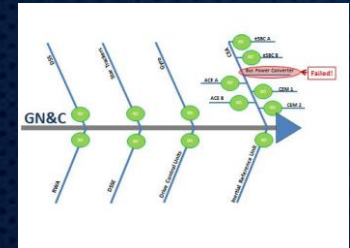




# But what about MBSE for . . . ?



- Failure Review and ADR
  - Capture of System Design for use in ADR
  - Potentially valuable, but . . .
  - Issues with visualization, compartmentalization (especially for shared payloads), maintenance in operations, etc.
- Mission Operations
  - Leverage of System Design for On-Orbit Handbook
  - Maintenance for evolution of system in ops (degraded performance, consumables, etc.)
- Risk Management
  - Integration as meta data in System Model
  - Reporting and visualization via modeling tool are relatively weak
  - Maintenance of information from another mechanism more ideal
- Others



Risk Rating = Likelihood x Severity

	1	2	3	4	5
Catastrophic	5	10	15	20	25
Significant	4	8	12	16	20
Medium	3	6	9	12	15
Low	2	4	6	8	10
Negligible	1	2	3	4	5

Legend:

- Catastrophic: STOP
- Significant: IMMEDIATE ACTION
- Medium: ACTION
- Low: MONITOR
- Negligible: NO ACTION

Likelihood: Improbable, Remote, Occasional, Probable, Frequent

# Challenges to Opportunities for MBSE



- Tools
  - SysML Implementations
  - Configuration Management (especially when digitally integrated)
- Integration with other Disciplines
  - Modeling?
  - Data Sets and Interoperability
- Communication
  - Customer
  - End-User
  - Inside Program
- Usability
- Model Organization & Management
  - Package Structure
  - One Model? Two? More?
  - Reference & Reuse