Experiences Deploying MBSE at NASA JPL

Frontiers in Model-Based Systems Engineering Workshop

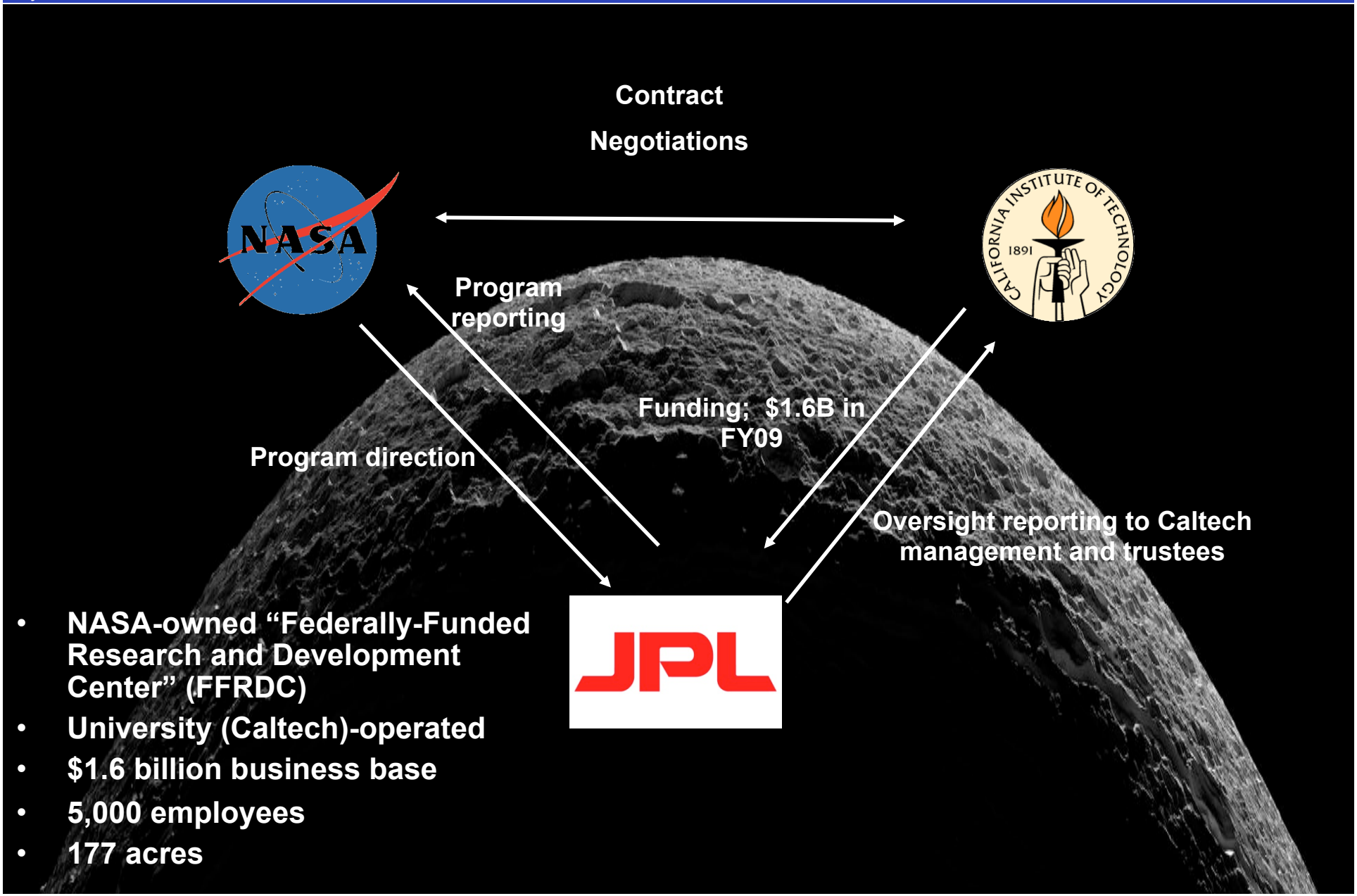
April 27-28, 2011
Georgia Institute of Technology

C. Lin
D. Nichols, H. Stone
S. Jenkins, T. Bayer, D. Dvorak

Jet Propulsion Laboratory, California Institute of Technology

JPL is part of NASA and Caltech

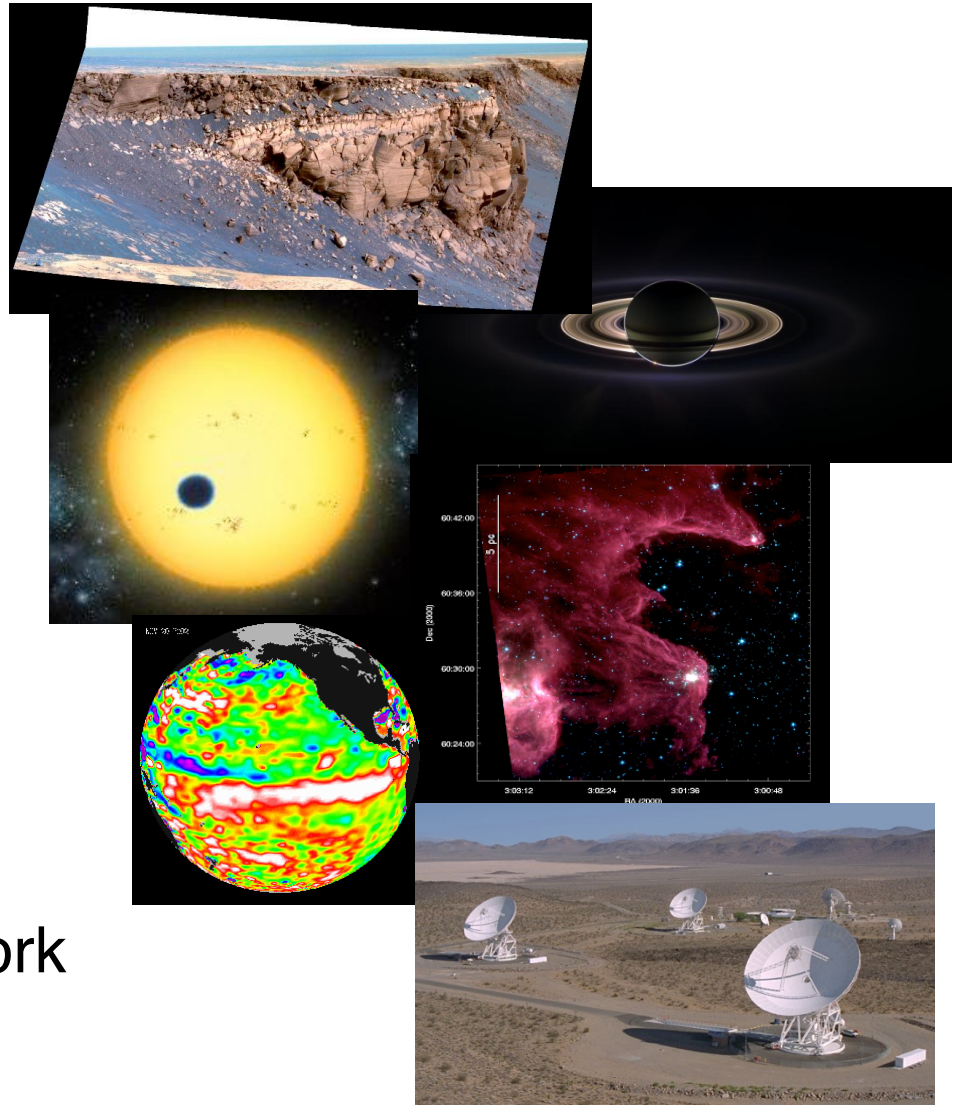
Systems and Software Division



- NASA-owned "Federally-Funded Research and Development Center" (FFRDC)
- University (Caltech)-operated
- $1.6 billion business base
- 5,000 employees
- 177 acres

JPL's Mission for NASA is *Robotic* Space Exploration

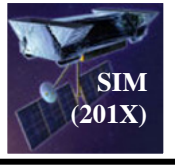
- Mars
- Solar system
- Exoplanets
- Astrophysics
- Earth Science
- Interplanetary network



JPL Space Missions

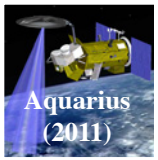
Systems and Software Division

Formulation (5)



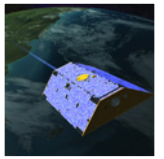
Planetary/Mars Astrophysics Earth Science

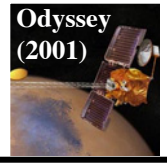
Development (5)



Planetary/Mars Astrophysics Earth Science

Operations (19)

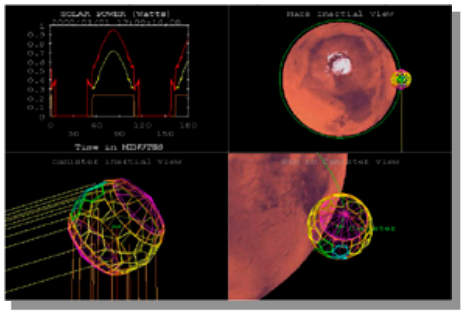




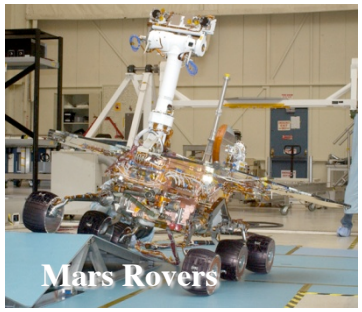
JPL Maintains an End-to-End Capability

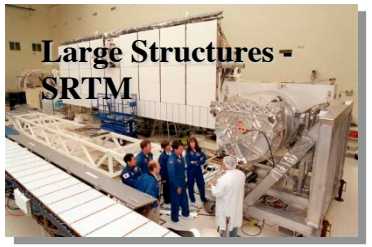
Systems and Software Division

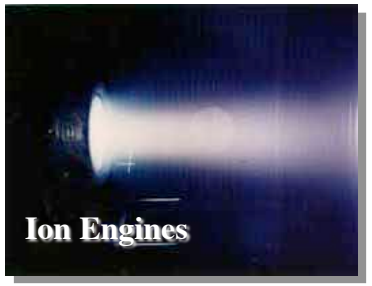
Project Formulation - Team X



Mission Design







Spacecraft Development

Integration and Test

Environmental Test

Real Time Operations

Presentation Roadmap

- How JPL started MBSE
- Systems engineering challenges
- The vision for model based systems engineering
- Infusion strategies
- Infusion challenges
- Lessons learned

Three Years Ago..

Frontiers in MBSE Workshop @GIT, 2008

Since then…

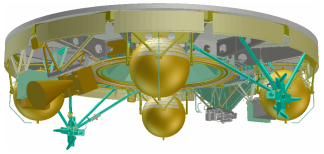
- JPL has increased its engagement in INCOSE and OMG
- Analyzed and articulated overarching SE issues and challenges
  - Connected the solution with the problem
- Developed an infusion strategy
  - Technical and cultural aspects
- Established a JPL MBSE initiative
- Engaged MBSE leaders in the field
  - e.g., Sandy Friedenthal, Chris Paredis, Russell Peak, Heinz Stoewer

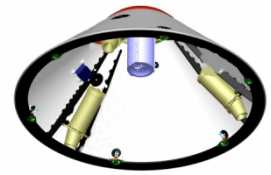
Challenges in Systems Engineering

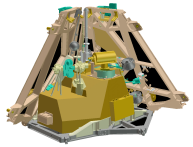
- Complexity of our systems continues to grow
  - More and more hardware functions are being replaced by software behaviors.
    - Yet hardware is becoming more like software
  - System level interactions (emergent system behaviors) are increasingly a major risk factor that cannot reliably be exposed by testing.
  - Natural-language based, single viewpoint specifications are inadequate to capture and expose system level interactions and characteristics.
  - System designs are spread across multiple documents
- Cost caps and competition motivate maximal reuse

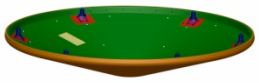
Mars Lander Spacecraft Exploded Views

MARS PATH FINDER (MPF)

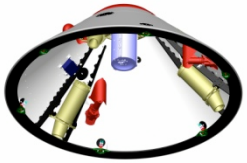


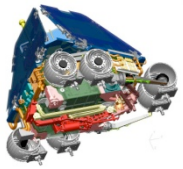


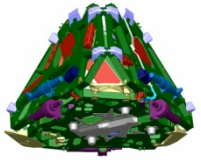


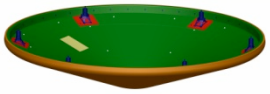


MARS EXPLORATION ROVER (MER)

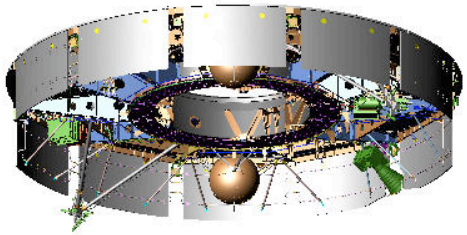


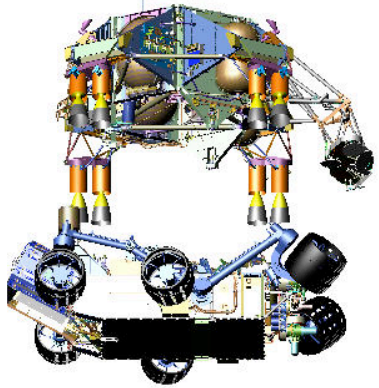






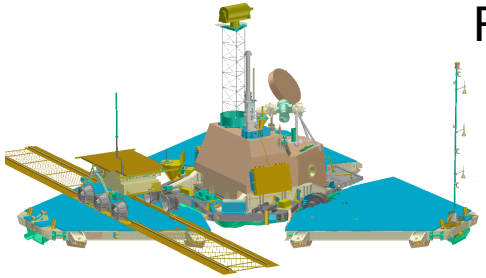
MARS SCIENCE LABORATORY(MSL)



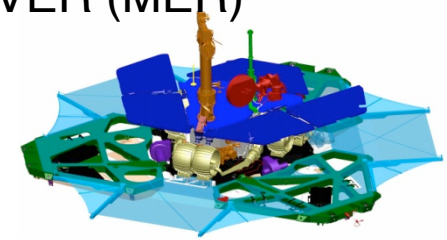


Mars Rovers Landing Mechanisms

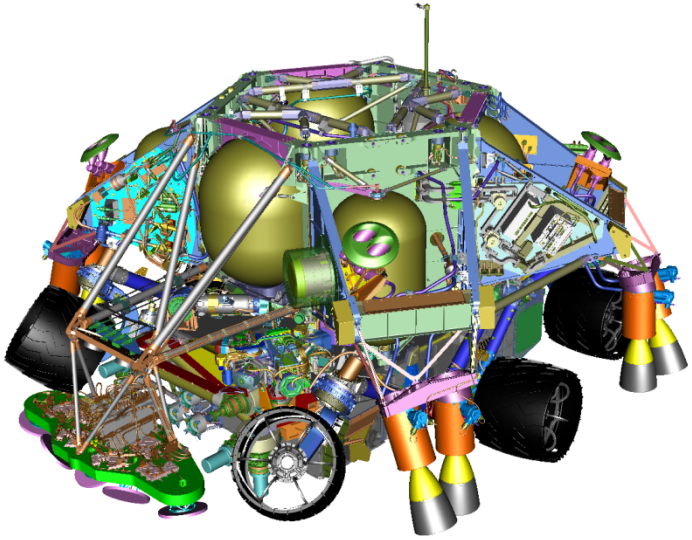
MARS PATH FINDER (MPF)



MARS EXPLORATION ROVER (MER)



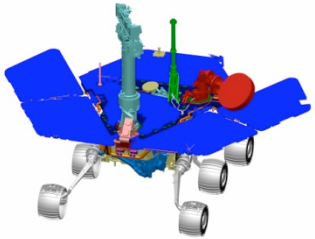
MARS SCIENCE LABORATORY (MSL)



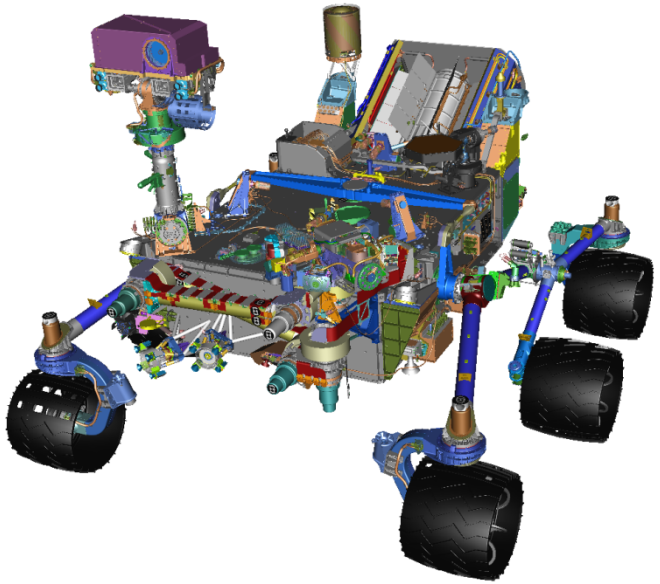
Mars Rovers Traversing

MARS PATH FINDER (MPF)

MARS EXPLORATION ROVER (MER)

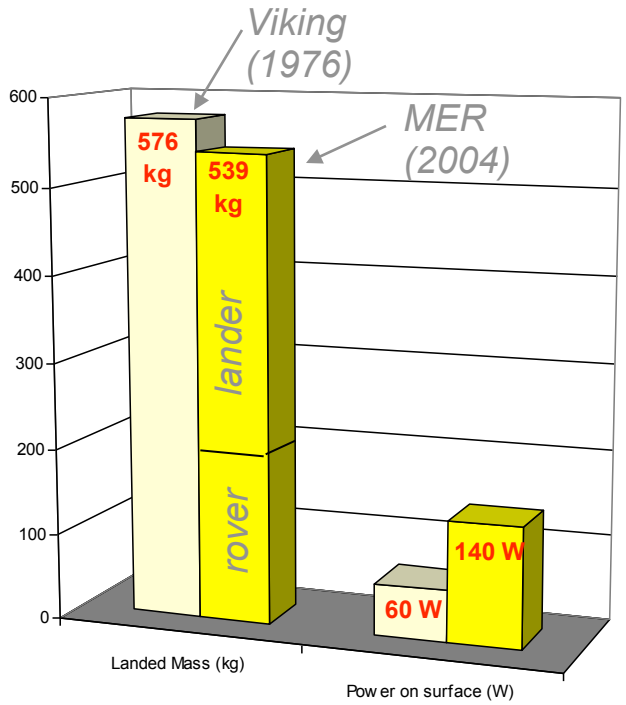


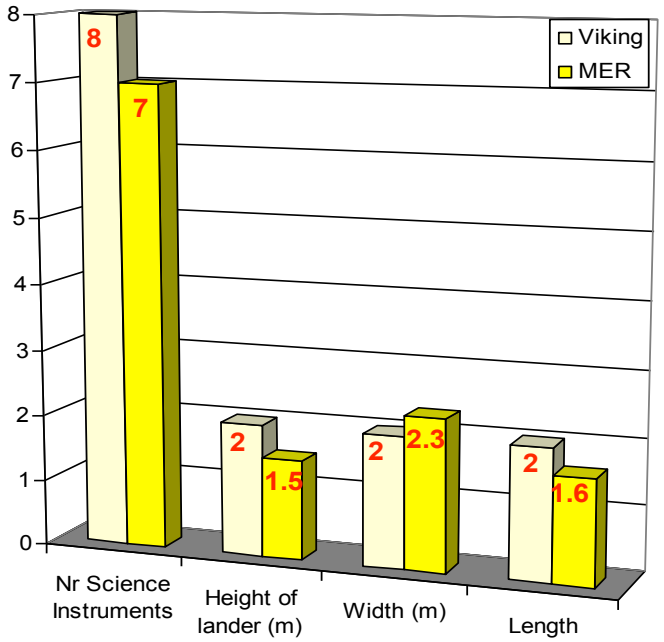
MARS SCIENCE LABORATORY (MSL)

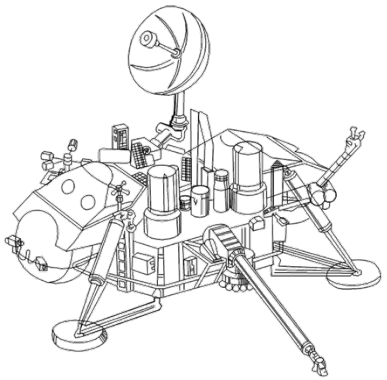


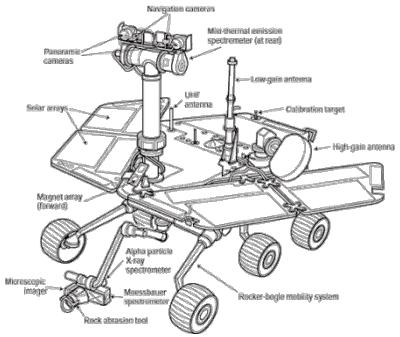
Comparison of Viking and MER: *hardware*

Systems and Software Division



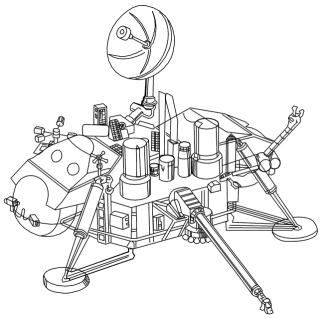


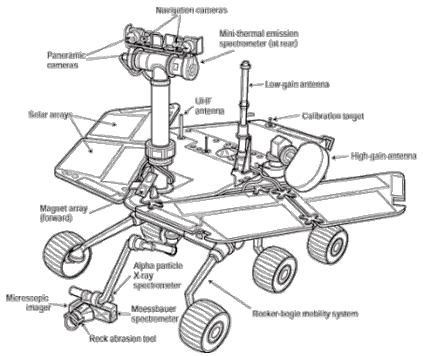


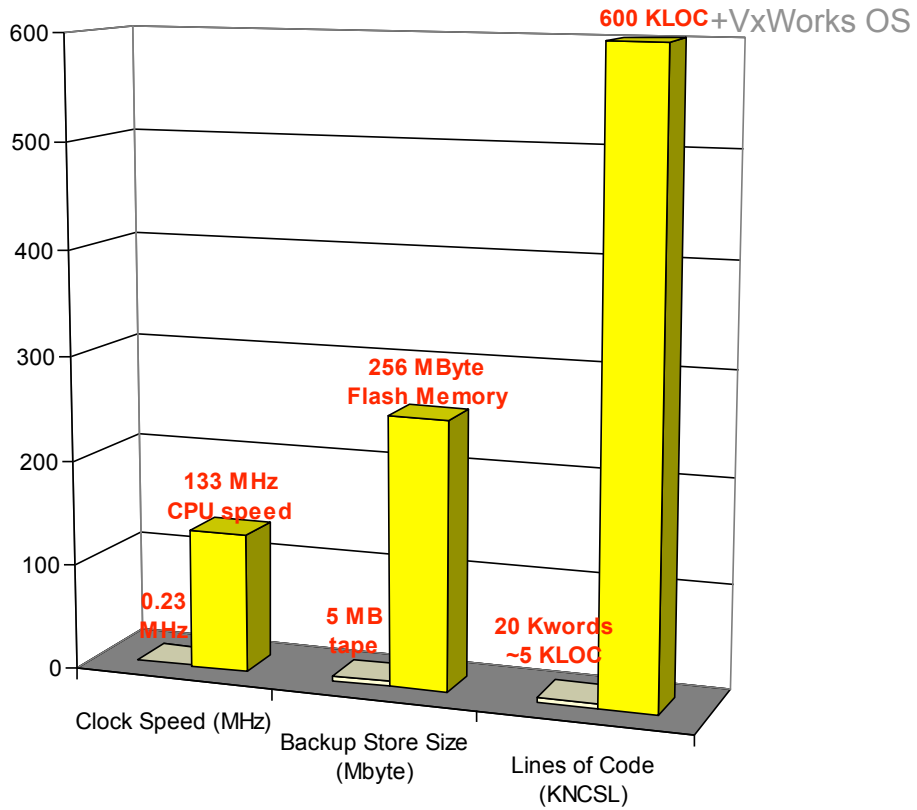


Comparison of Viking and MER: *software*

Systems and Software Division







~500x faster CPU

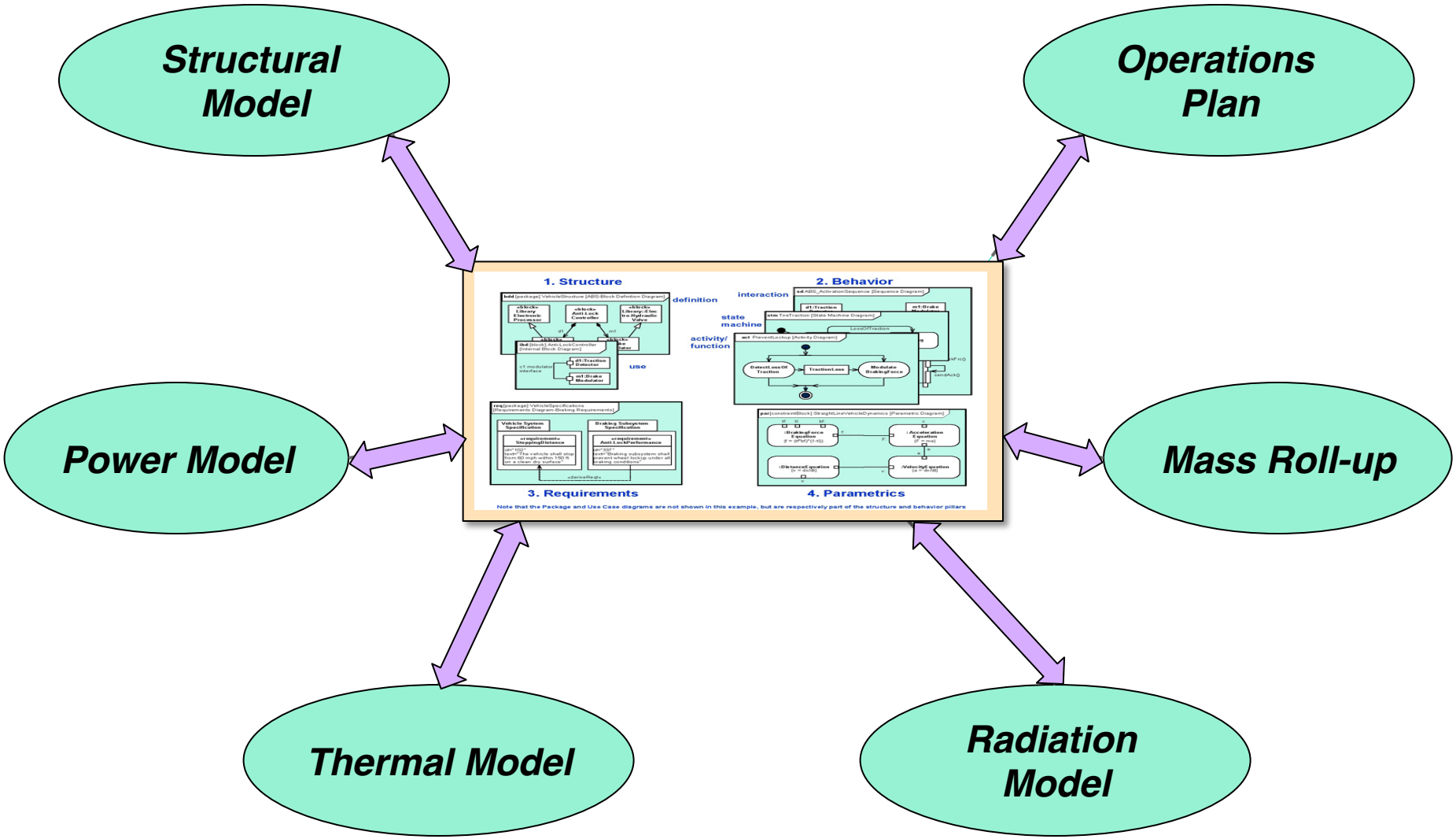
~50x more backup storage

>>100x more code (size of VxWorks operating system *not counted* in graph)

~0.3x sw development time (540 man-months for MER / 1609 man-months for Viking landers)

…the shift: systems engineering with models

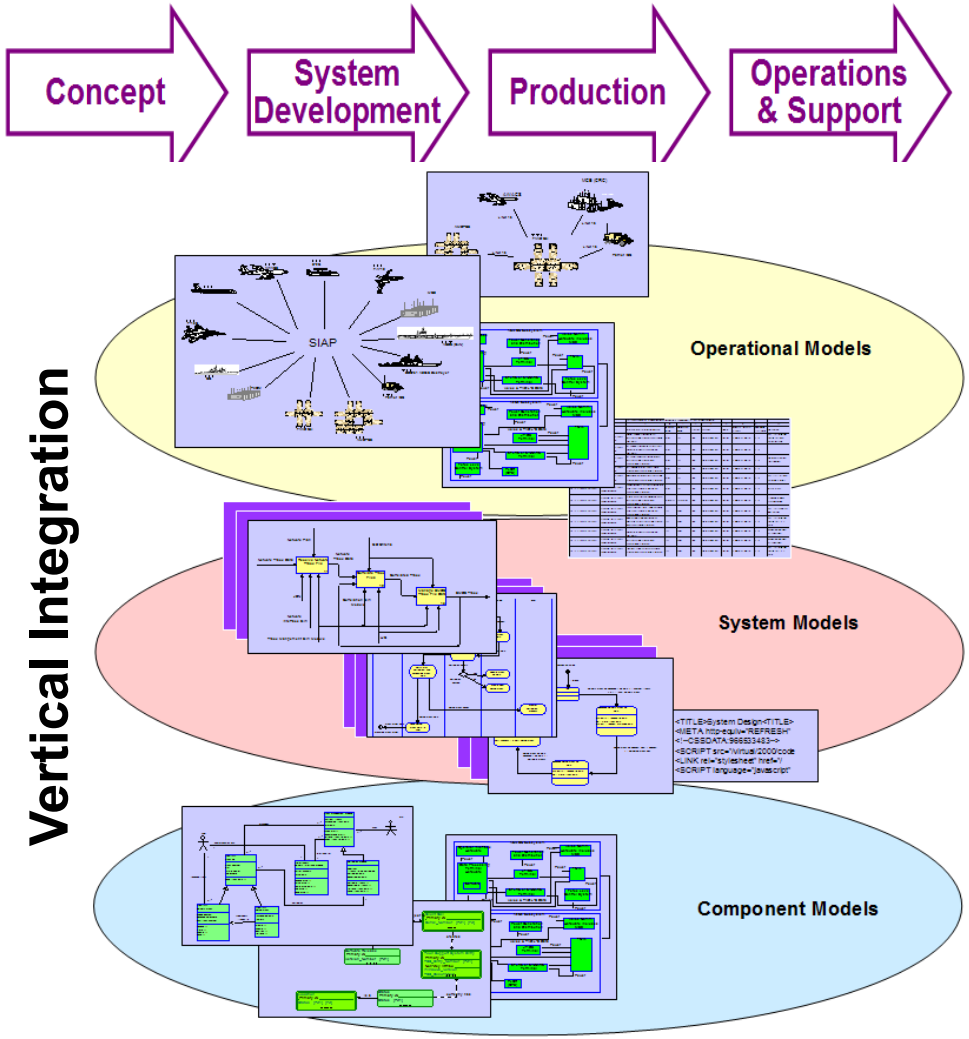
Current State → Future State



Integrated Model-Centric Engineering

Integration Across Life-Cycle and Domains

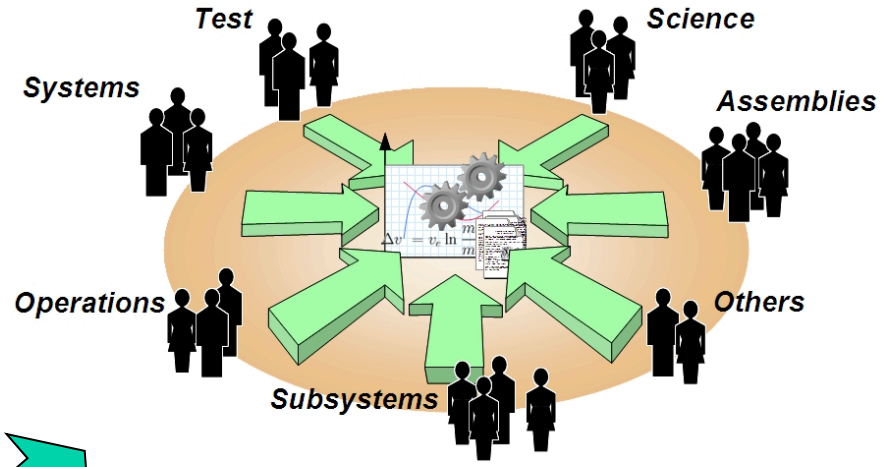
- **Formalizes the practice of systems engineering through the use of models**
- **Broad in scope**
  - Integrates with multiple modeling domains across life cycle
- **Results in quality/productivity improvements & lower risk**
  - Rigor and precision
  - Communications among system/ project stakeholders
  - Management of complexity
  - Design capture enhances reusability



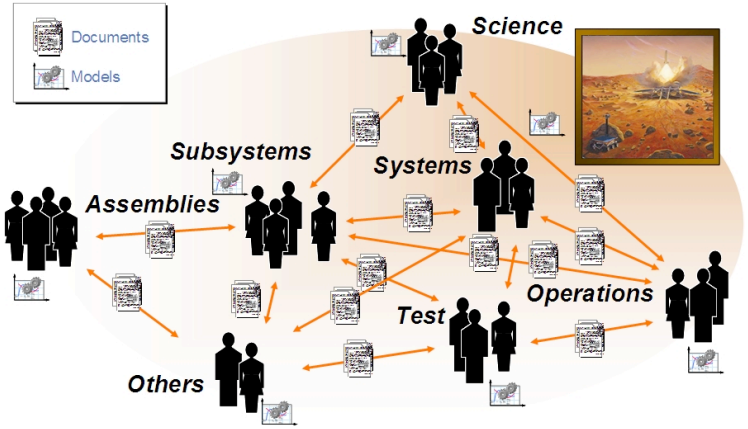
JPL An Integrated Model-Centric Engineering Vision at JPL

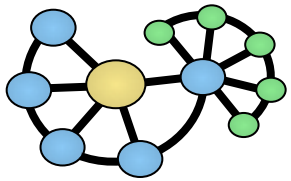
Systems and Software Division

Mission: To advance from our current document-centric engineering practices to one in which structural, behavioral, physics and simulation-based models representing the technical designs are integrated and evolve throughout the life-cycle, supporting trade studies, design verification and system V&V



Future: Reusable model-driven with integration & simulation capability



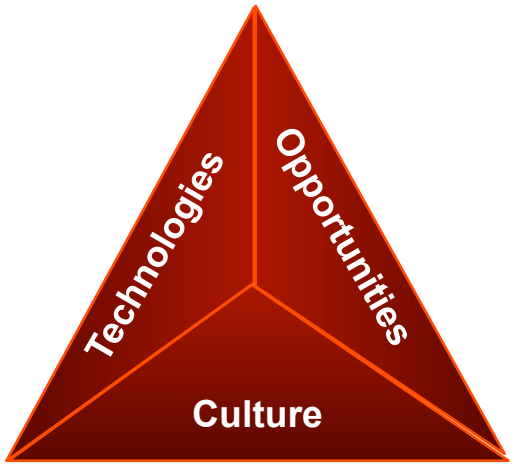


Today: Document driven & standalone models

Integrated Model-Centric Engineering (IMCE)

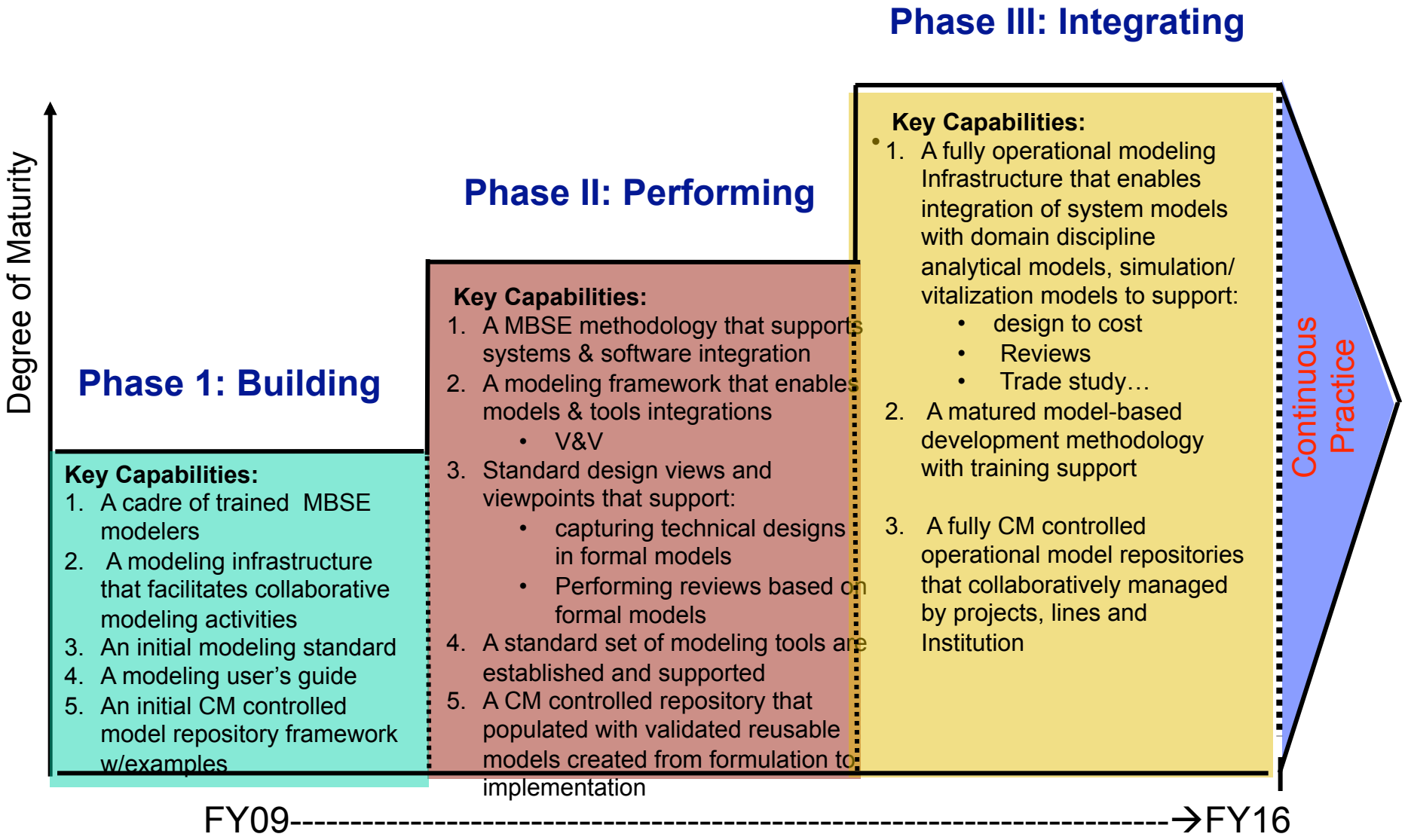
- **IMCE** is the **JPL initiative** that helps accelerate the application of a model-based engineering style to the integration of system designs at various level of detail across the full project life cycle

- Such an initiative needs to consider and address…



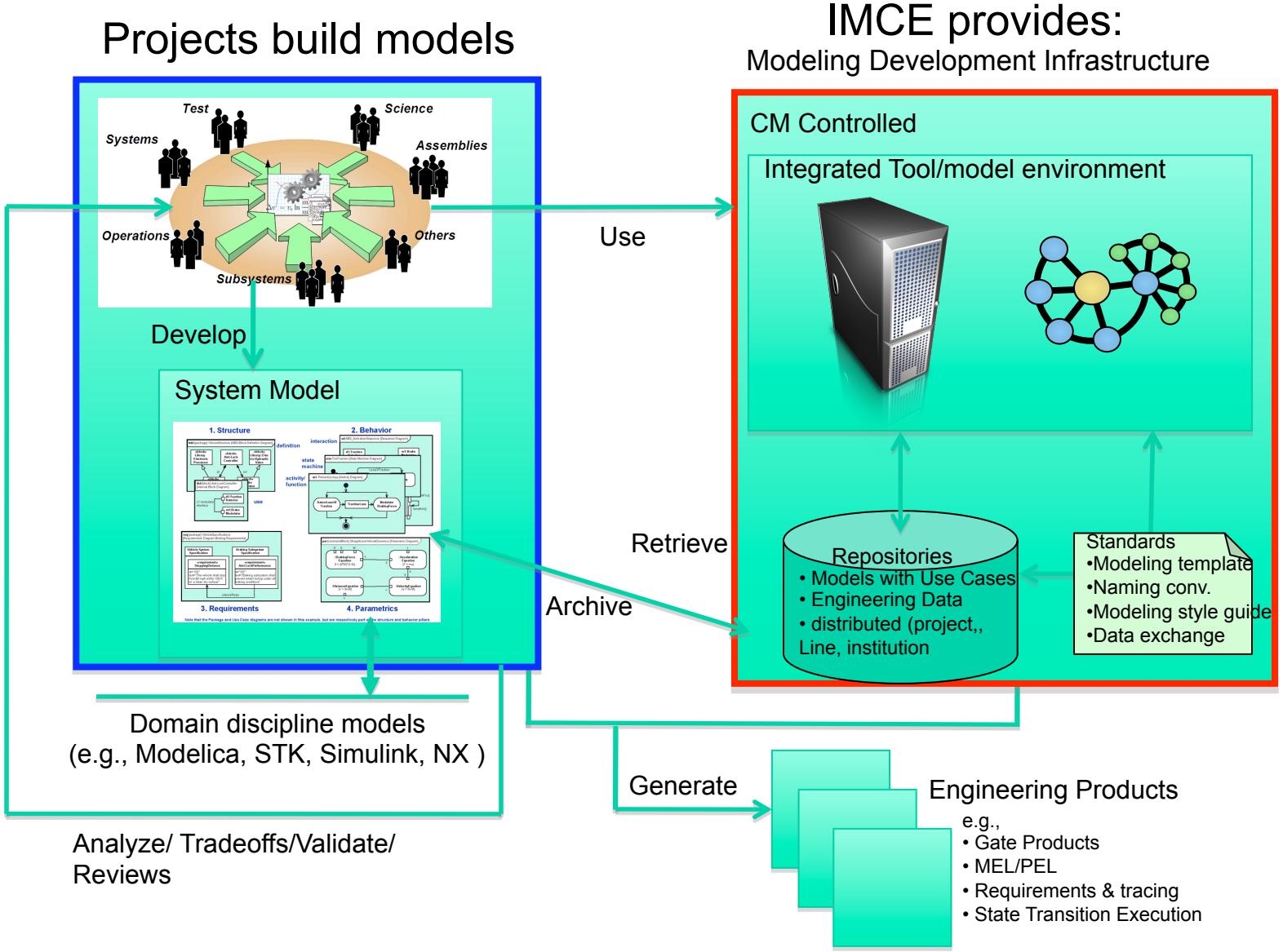
Multi-Year Strategy: Capabilities

Setting Expectations: Such an effort takes time to achieve

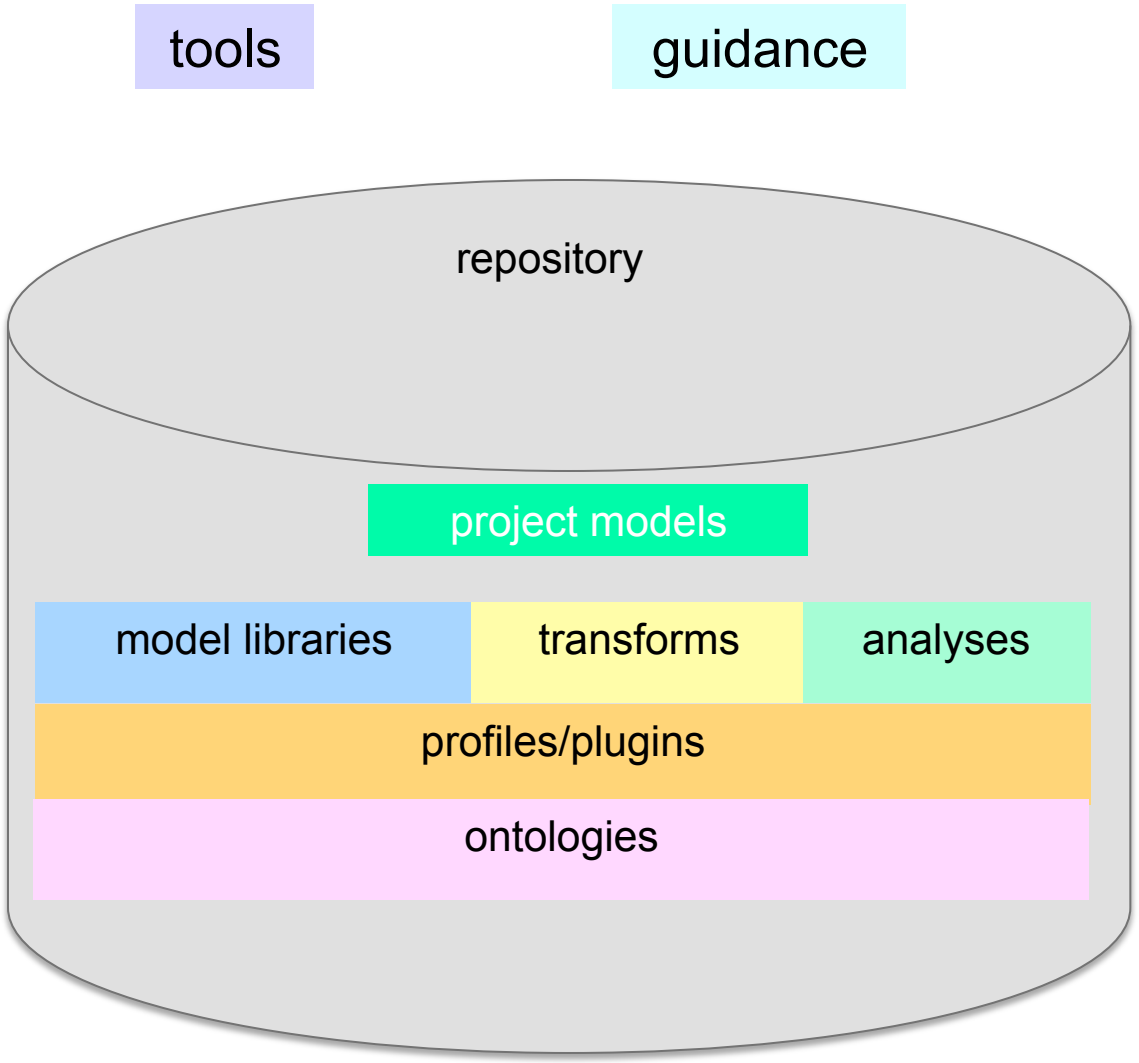


Integrated Model-Centric Engineering: Ops concept

Systems and Software Division



Essential Modeling Elements Architecture



…the strategy

Infusion Strategy

It’s the results

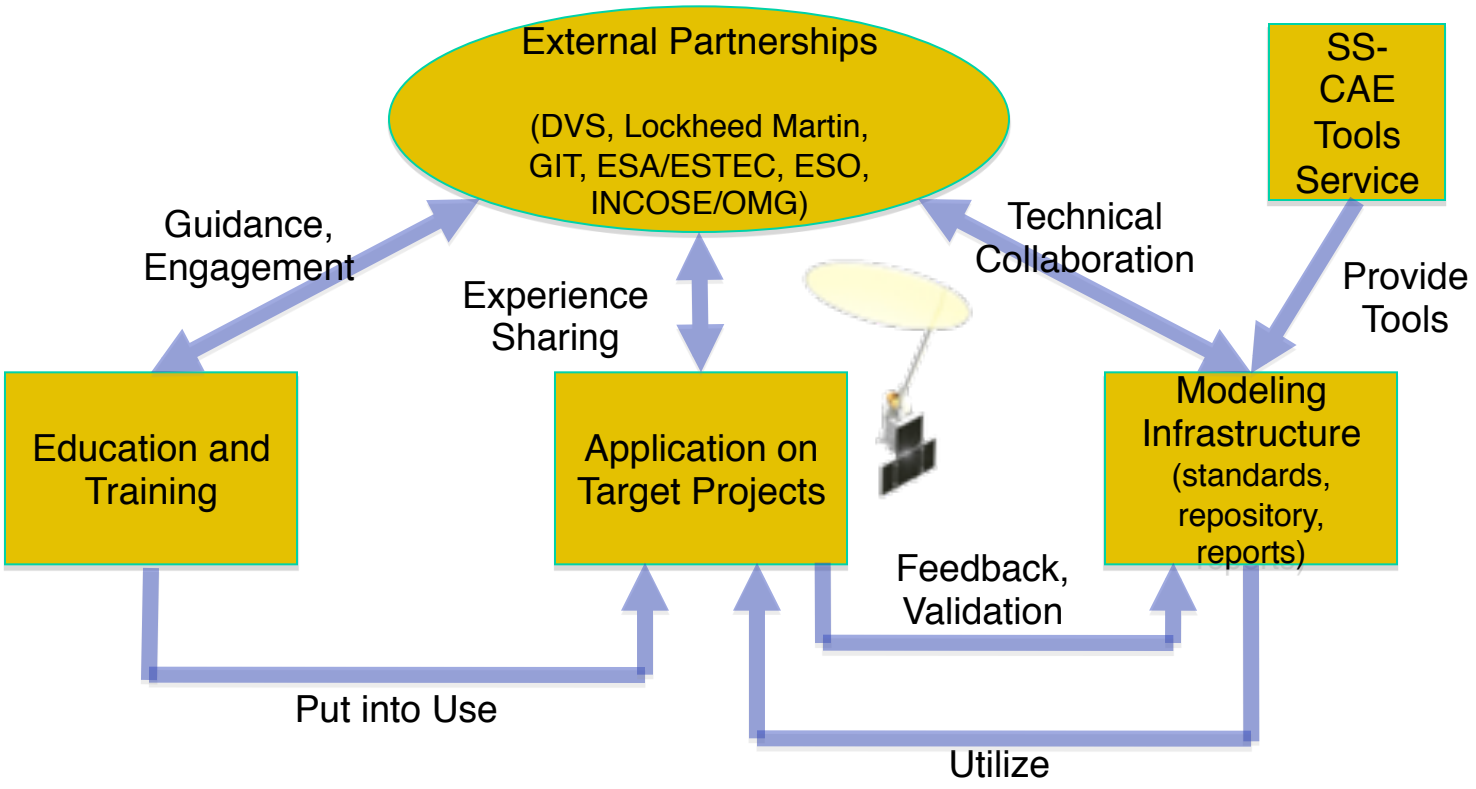
It’s the support

It’s the people

Multi-pronged approach

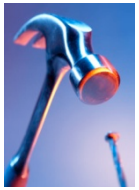
- Educate and train an initial cadre of modelers
  - Pair domain experts with early career hires
- Build on grass-roots efforts
- Work with strong advocates and advisors consisting of international and national experts
- Establish an institutionally-supported modeling environment
  - Define modeling standards, enable collaborative modeling effort, build a reusable model repository; provide support to system model developers
- Identify and build applications
  - Develop system models that have immediate benefits to project's needs
  - Put training into practice
  - Modelers partner with project's system engineers to get early buy in
  - Use the initial application to validate the reusable modeling environment
- Partner with industry, INCOSE, academia and OMG to learn, contribute and stay current

Interactions Among Key Initiative Elements



Supporting Change

- Carrot
  - Rewards and recognition



- Stick
  - Integrate into standard practices

- Evangelism
  - A persistent and consistent message from management
  - Provides awareness and distinctions

- The X-Team* approach
  - Go “outside” – make external outreach a *modus operandi* from day 1

* *X-Teams: How To Build Teams that Lead, Innovate, and Succeed* (Harvard Business School Press, 2007)

…some results

Highlights of Accomplishments

- Released IMCE Concept of Operations for two target audiences
  - Developers of IMCE capabilities
  - Users of IMCE capabilities and MBSE adopters
- Established IMCE Technical Advisory Board
  - A body of external and internal experts in SE and MBSE
- Trained over 120+ engineers (across all disciplines); 50+ active practitioners
- Formed a modeling early adopter group; 80+ members; meet twice week
  - Led by early career hires

Highlights of Accomplishments (Cont.)

- Developed JPL-specific ontology and SysML modeling profiles for space mission specific applications
  - Essential for modeling initiation, model reuse and integration
  - Consists of three architecture layers: Foundation, Discipline and Application
    - Peer reviewed by system engineering leaders; adopted by JEO

Application Level:
Star tracker, Antenna, Thruster, Sun sensor …

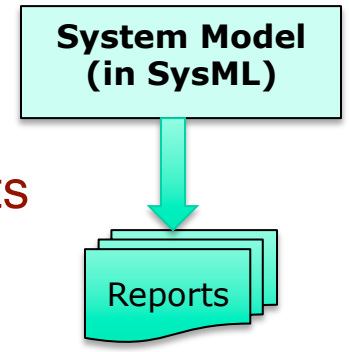
Discipline Level:
Mechanical, Electrical, Thermal, ACS, NAV …

Foundation Level:
Base, Mission, Project, Physics, Units …

- Established an modeling tool environment
  - Provided by the JPL Institutional tool service
  - Consists of MagicDraw and Teamwork products

- Producing an initial SysML modeling guide
  - JPL specific, with examples

- Developed and demonstrated techniques for producing reports directly from SysML models



Highlights of Accomplishments (Cont.)

- Completed three summer projects that focused on models/ tools integration
  - Summer students from Penn State, GIT, UCLA
  - Mentored by IMCE staff and experts from telecom, mechanical and instrument engineering domains
    - Electra (UHF Relay Transceiver) model
    - Model Transformations
    - SIM Model Interchange between SysML and NX
    - Use SysML and electrical ontology for a senior project

- Project Adaptations
  - Established a strong collaboration with the Jupiter Europa Obiter (JEO) project system engineering team
    - First project to adopt MBE from the start
  - Over the last two years more than 6 projects have applied SysML to model part of system under development

Project Applications to Date

MBSE Activity	Short Description
JEO* Architecture Framework	Web App for model based mission architecture.
Dawn GDS	Modeling GDS Configuration
MSL GDS	Modeling GDS Configuration
MSL System Modeling	Modeling system behavior models and interfaces
SIM	Capture instrument models baseline design and output parameters to other models
LISA	Modeling Lisa Flight System and Micro-Thruster
SMAP	Model flight SW architecture Architecture review focused on model structures
MGSS	Model based techniques and languages to re-architect Mission Operations processes
OPALS	Integration and Test and System V&V
SAVIO/Constellation	Captured baseline design
Cielo/SysML integration	Exploring the integration of a finite element model and simulation into SysML

*Proposed missions

…some experiences and lessons learned

Barriers To Infusion

…the chicken or the egg

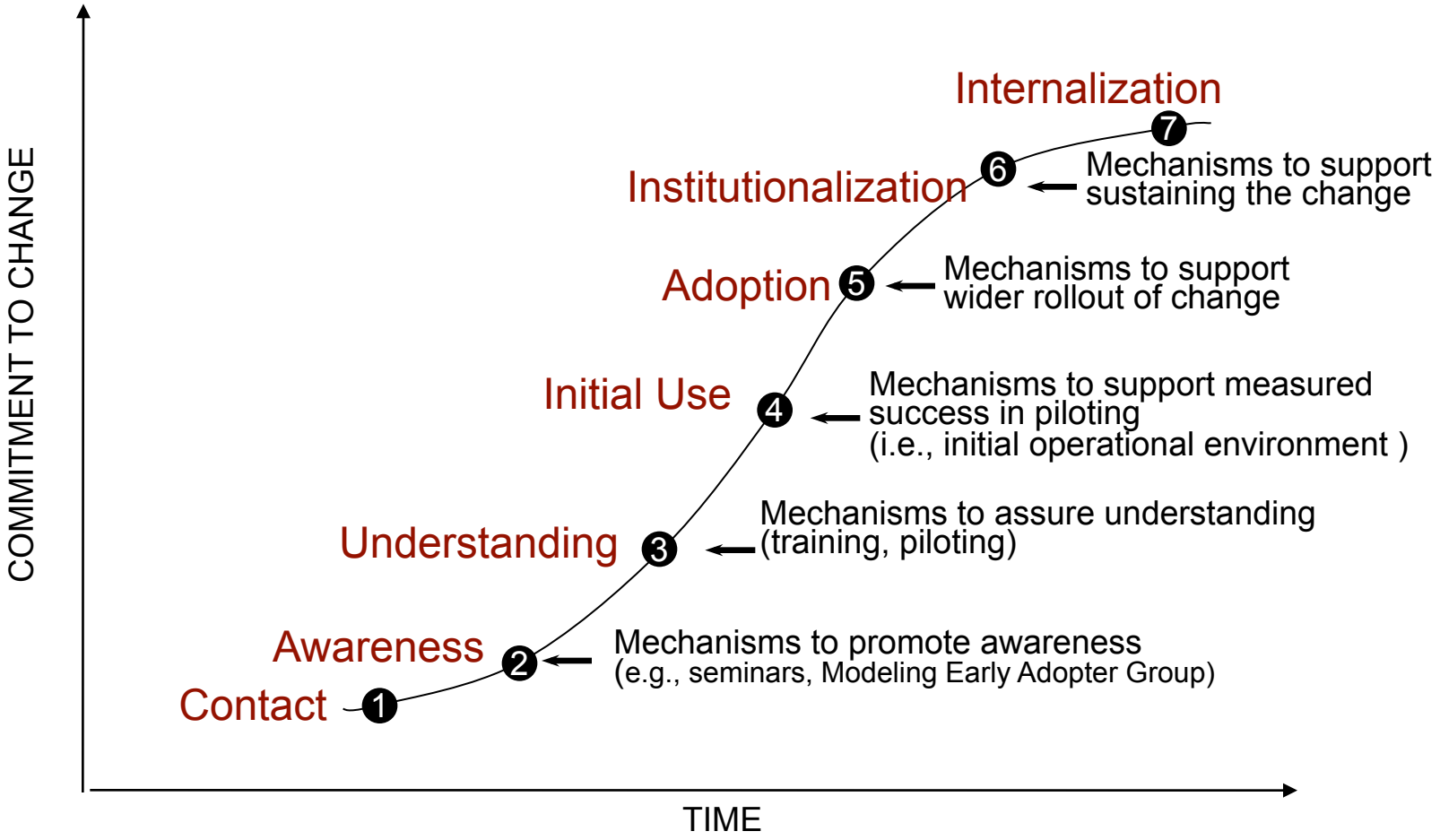
"…a decision is an invitation to debate"

"…better is the enemy of good enough"

Lessons Learned: Strategy

- Young engineers are arriving already versed in MBSE…
  - the transition to MBSE is going to happen whether we help or not
  - pair young engineers with gray beards
- Collaboration has been essential (industry, other space agencies, academia)
- Infusion can be gradual, both in time, and in project space
  - benefits are evident even when systems modeling is used in a modest way on a single subsystem
  - the simple act of creating a formalism is by itself a significant help in communication and understanding
- Culture change does not follow a project life-cycle

Lessons Learned: Understand the importance of Cultural Change Management



Adapted from Out from Dependency: Thriving as an Insurgent in a Sometimes Hostile Environment, SuZ Garcia and Chuck Myers, SEPG Conference, 2001

Lessons Learned: Just Modeling Isn't Enough

…we also have to agree on…

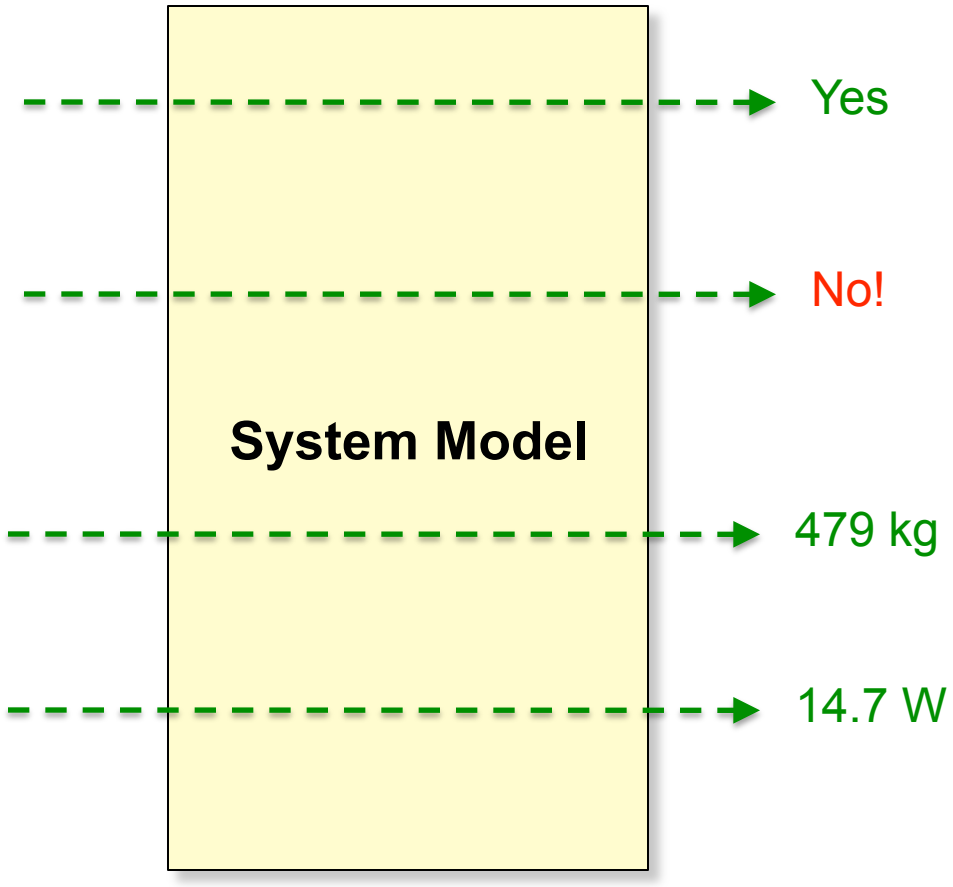
- Ontology
  - What concepts are important to us?
  - What properties and relationships do those concepts have?
  - How do we name concepts and properties?
- Notation and exchange syntax
- Tools and model repository
  - How do I create models?
  - Where can I store my models?
  - Where can I find other models and relate to their content?
- Model data access mechanisms
- Model validation rules and constraints
- Configuration management procedures
- Relationships between engineering deliverables and model content

…we model to get work done…

Lessons Learned: The Power of Modeling

Imagine that a system model answers queries such as …

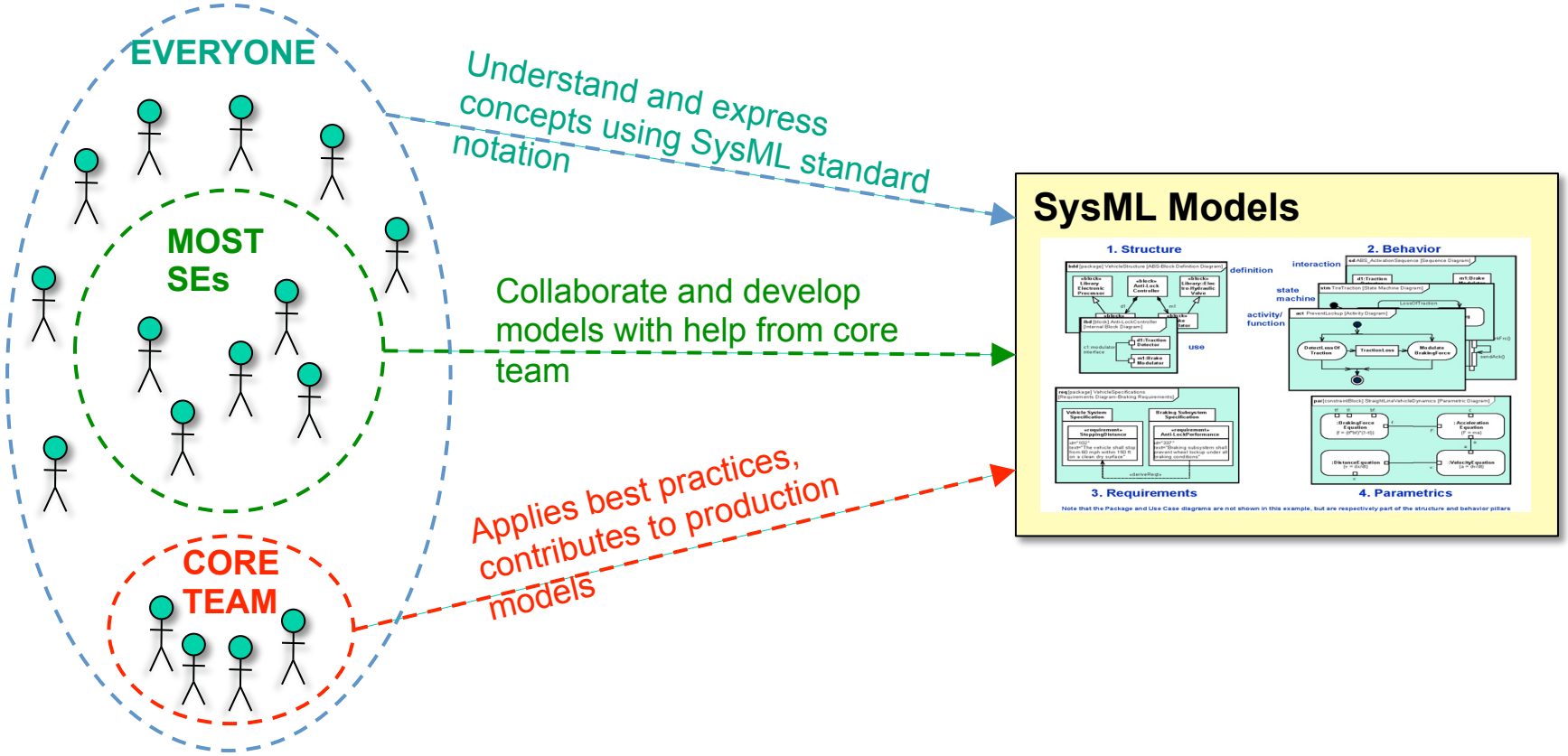
- Does every component trace to a requirement?
- Have both sides of every interface been specified?
- What is the dry mass of the flight system?
- What is the maximum load on power bus 3?



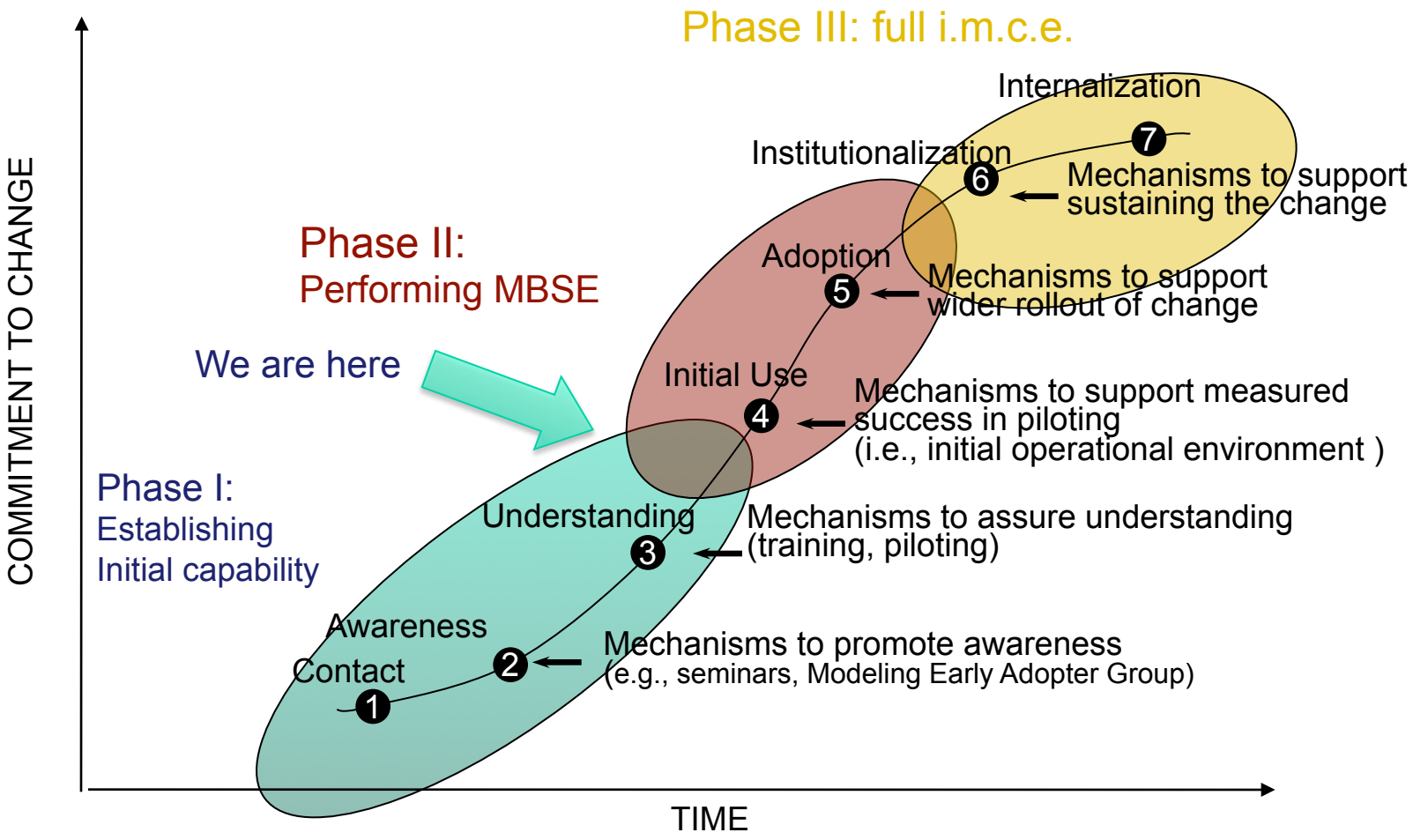
A system model makes this possible.

Lessons Learned: SysML and Tools

Roles: Contributors, Model Editors, and the System Model



The IMCE Cultural Change Roadmap



Adapted from Out from Dependency: Thriving as an Insurgent in a Sometimes Hostile Environment, SuZ Garcia and Chuck Myers, SEPG Conference, 2001

Summary

- Management of complexity and cost-effectiveness drive the need for MBSE
- We are finding real tangible benefits in early applications
- A multi-pronged, evolutionary infusion strategy is the most effective in the JPL environment
- Remember: MBSE is SYSTEMS ENGINEERING, not modeling

Thank You!