



# Model-Based Architecting

Presentation to AIAA Space 2012/CASE

Brian Muirhead  
Jet Propulsion Laboratory  
California Institute of Technology

# Agenda

---

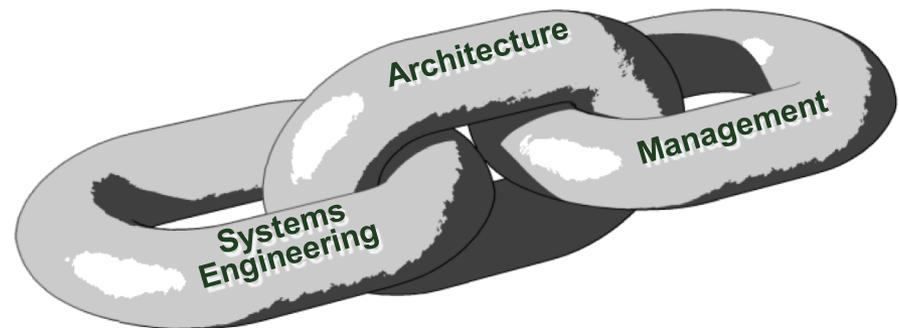


- ◆ **What is Architecting and Architecture?**
- ◆ **Components of Architecture**
- ◆ **Principles**
- ◆ **Architecture Framework**
- ◆ **Transition to Model-based Architecting**
- ◆ **Path Forward**
- ◆ **Conclusions**

# What is Architecture?

---

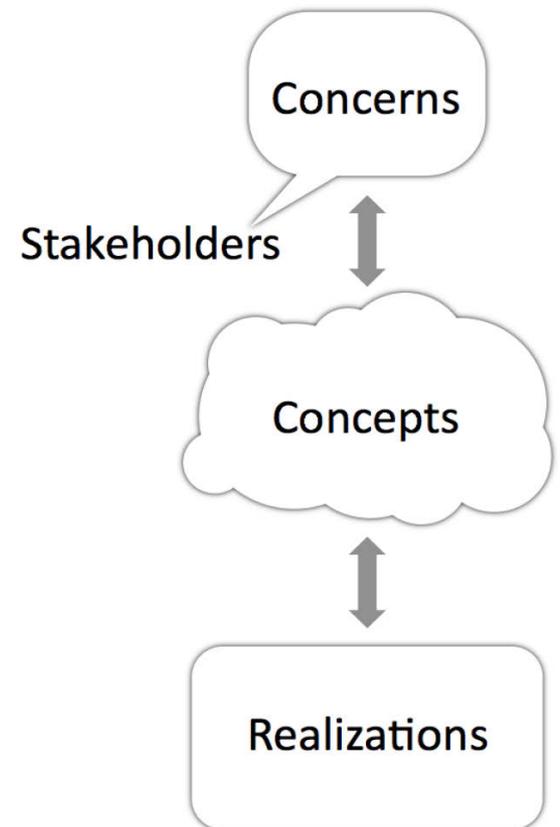
- ◆ **Architecture is the fundamental organization of a system, embodied in its components, their relationships to each other and the environment and the principles governing its design and evolution (IEEE Std 1471-2000)**
  - **Links Systems Engineering & Management by balancing technical and programmatic considerations**
- ◆ **Architectures are fractal. One person's architecture is another person's detail. One person's system is another person's component.**



# Cardinal Rule of Architecting



- ◆ **Architecture is an *understanding* of the sort of system that needs to be built and our expectations for it, with all their compromise.**
- ◆ **Product of architecting is not a system!**
- ◆ **As much a matter of *why* and *how* as of *what*.**
- ◆ **It underlies the designs ability to meet objectives/constraints and satisfy stakeholders**
- ◆ **A design is the embodiment of an architecture. Designs address what is to be built and how**
- ◆ **As systems become more complex the need for an effective architecting effort becomes essential**

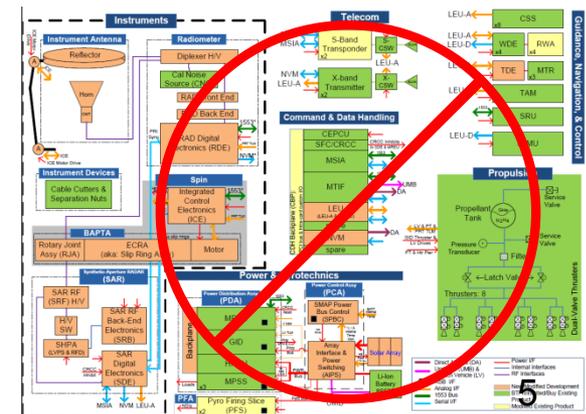


**Concepts** are essential abstract ideas, formulated to address **Concerns**, but taking implementable form in **Realizations**.



# What Architecture Is Not!!

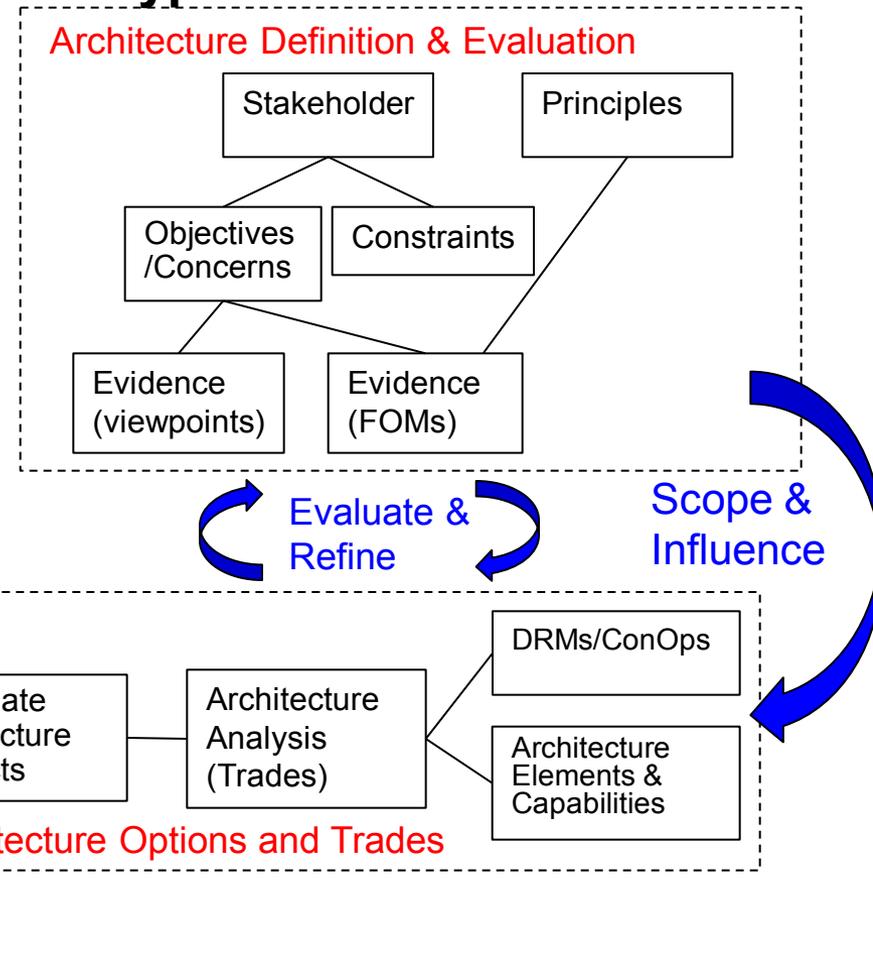
- ◆ Architecture is not a broad brush effort confined to early development
  - Dictates what possibilities are allowed, while still remaining faithful to stable concepts selected to fulfill system objectives
- ◆ Architecture is not opaque pictures, block diagrams, lists, or other schematic representations of the design
- ◆ Architecture is not requirements
  - Architecture provides the rationale for requirements
- ◆ Architecture is not fickle, or subject to routine refinement
  - Architecture provides a stabilizing influence through its well-considered form, expectations, rules, and attention to fundamentals



# Components of Architecting

## ◆ Architecture takes into account several types of information

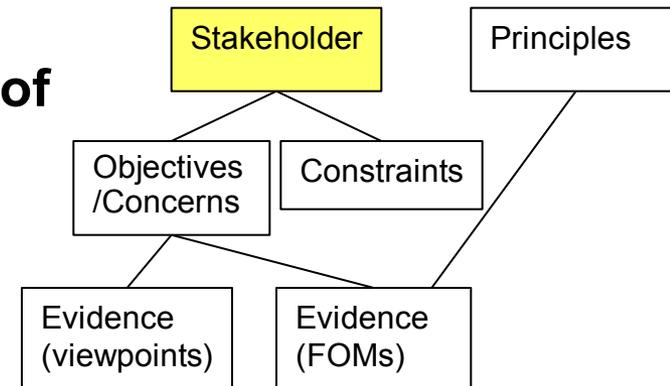
- Stakeholders
- Stakeholder Objectives/concerns
- Constraints
- Architecture Principles
- Figures of Merit
- Evidence – viewpoints & FOMs
- Design Reference Missions/OpsCons
- Elements and capabilities
- Architecture Trades/Analysis
- Candidate Architectures
- Decisions
- Requirements.



**Architecting is the art of balancing these elements (including as they evolve/change) to maximize the stakeholder satisfaction**

◆ **Influential people or groups who have something to gain or lose in the selection of architecture options.**

- Essential for the full complement of stakeholders to be identified and engaged in the architecture process.
- Influence can be drawn from many sources (legal, financial, advocacy, political, etc.)
- Not merely titles, groups or organizations, but an actual person that the architect can engage and who is authorized to negotiate for the community they represent.



**A complete set of stakeholders will represent widely different communities, all of whom have the potential to significantly impact the success of your program**



# Stakeholder Objectives/Concerns & Constraints

◆ **Objectives/Concerns are the criteria by which the success of an architecture is measured**

- Architectures are never better than the quality of their objectives and how well they capture the stakeholder's concerns.

◆ **Requires a clear definition of:**

- Stakeholder's Objectives and/or Concerns:

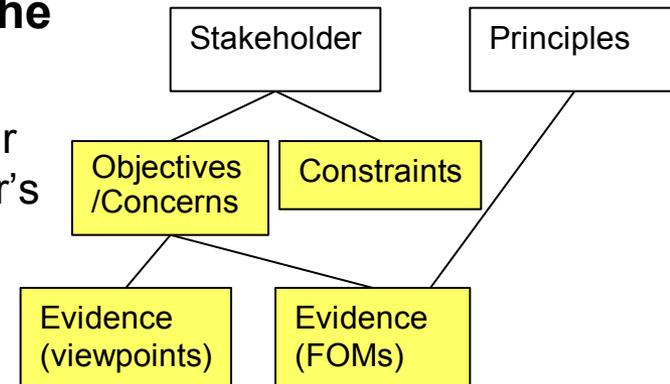
- Whats: Performance, functionality, quality, cost, reliability...
- Hows, e.g. how the system comes together, or is operated, or relates to other developments:

- Stakeholder-owned Constraints that restrict the valid architecture solutions

- An agreed-upon set of actionable and objective Evidence that will be used to show the stakeholder how the architecture addresses their inputs

- **Viewpoints:** budget, mission timeline, partnership allocations
- **Figures of Merit (FOMs):** affordability (DDT&E, recurring), sustainability (ability to engage public, science return, etc.)

◆ **Architects job is to help stakeholders express all of these concretely.**



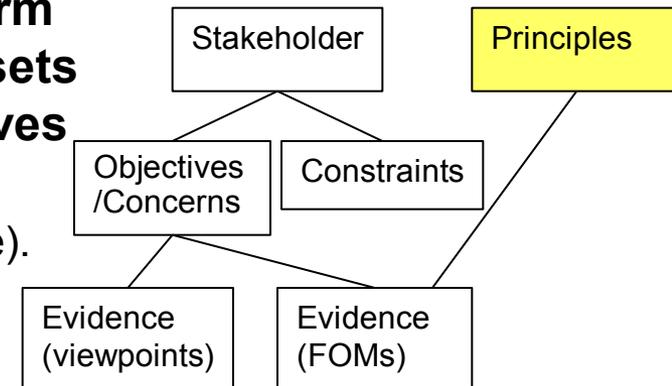
**Stakeholder inputs are hard to gather, and will be diverse, conflicting and dynamic. Balance and compromise are essential to success.**



# Architecture Principles

◆ **The fundamental rules and guidelines that inform and support the way in which an organization sets about fulfilling its business & technical objectives**

- Fundamental concepts used to help make decisions on key trades (can have a big impact on architecture).
- Often distilled from previous experiences
  - Success ← Wisdom ← Experience ← Mistakes
- Examples: TLYF-FLYT, Maintain adequate margin, routine cadence of missions, Inspire through numerous “firsts”, leverage commonality, etc.



◆ **A good principle is... what you really care about**

- Well substantiated
- Clear about applicability and application,
- Relatively easy to explain, and
- Stable (slow to change)

◆ **Principles foster order, structure, elegance**

- Commitment to fundamentals
- Basis for architectural integrity

**A good principle is worth fighting for, and the last thing you're willing to give up.**

# General Principles of Architecture

---



- ◆ **Developed to promote an early and sustained consensus of system understanding from many technical and programmatic points of view**
- ◆ **Elegant architecture demands attention to purpose, understandability, and robustness of the system as a whole, even as the details develop**
- ◆ **Works best as an overtly distinct and sustained effort**
- ◆ **Good architecture provides stable guidance to development by getting the fundamentals right and asserting them steadily and consistently**
- ◆ **The integrity of a system lies, not merely in passing tests, but in demonstrated adherence to solid architectural assertions/principles**
- ◆ **Proper architecting makes inherently complex systems understood and manageable. “Complexity is a measure of how well a system is understood by those concerned with its development and use.”**

# Content Principles

---



- ◆ **Usefulness and acceptability lie in the association of systems with stakeholder concerns, so architecture must cover both, and do so thoroughly and uniformly**
- ◆ **Architecting defines a space of designs by allowing for variation (margins, contingencies, etc.) and defining what variations are allowed (rules, patterns, etc.) across the full spectrum of concerns**
- ◆ **Good separation of concerns requires overt attention to concepts, as a way to ensure that all issues get proper attention, and to map concerns to realization in an understandable and modelable way**
- ◆ **Conceptual and realizational efforts should work in partnership to achieve conceptual integrity and support realization requirements with robust rationales**
- ◆ **When all concerns are addressed in collaboration, and mapped through well-separated concepts to realization, architectural layering is much cleaner and easier to understand**

# Example Principles and Rationale

---



- ◆ **Architecture must be consistent with the business model**
  - Past architectures have failed because they took an unrealistic approach to accommodating the business model
- ◆ **Be robust to changes in stakeholder support and priorities**
  - Need flexibility and/or inertia to accommodate inevitable changes in stakeholder support and/or objectives/constraints
- ◆ **Allow for engagement of international partners (IPs) across the architecture**
  - IPs come with broadening capabilities and their own resources, provide broader political support (inertia) and can mitigate some programmatic risks
- ◆ **Use an evolutionary development strategy**
  - Allows for incorporation of advancing technologies
  - Provides for more rapid and periodic “firsts”
  - Helps deal with budget limitations/fluctuations

# Interesting and Useful Heuristics

---



From Maier and Rechtin, *The Art of Systems Architecting*, CRC Press 2009

- ◆ **Success is defined by the beholder, not by the architect**
- ◆ **Don't confuse good architecture description with good architecture**
- ◆ **Politics, not technology, sets the limits of what technology is allowed to achieve**
- ◆ **Cost rules**
- ◆ **A strong, coherent constituency is essential**
- ◆ **Technical problems become political problems**
- ◆ **The best engineering solutions are not necessarily the best political solutions**



# Architectural Framework Elements

---



- ◆ **Stakeholders determine whether a system is useful and acceptable. It is typically necessary to tailor interactions (information and communications) uniquely for each stakeholder.**
- ◆ **Concerns express the interests of stakeholders: objectives, constraints, quality and performance attributes, and other criteria, both technical and programmatic. Architectures reflect the reconciliation of concerns in negotiated, mutually consistent success criteria.**
- ◆ **Views and Viewpoints are the means by which an architecture is communicated to stakeholders, developers, and operators. A view is a tailored description that narrowly addresses in a defined way the interests of a particular viewpoint.**
- ◆ **Model-based architecting provides the ability to more formally connect these notions in order to better see what views share, and thus how concerns compete**

# Architectural Framework Fundamentals (cont.)

---



- ◆ **Models**, are depictions or descriptions that address some aspect of the architected system or its behavior.
- ◆ **Analyses and Scenarios** are necessary in many views. An analysis is any consideration of the models that draws some conclusion about them.
- ◆ **Trades (or Decisions or Rationales)** are the essential link between stakeholder concerns and other architectural features, explaining why an architecture is the way it is
- ◆ **Principles (or Patterns or Guidelines)**, should be explicitly collected and justified so that the decisions they guide can be explained in a form communicable to subsequent efforts
- ◆ **Elements and Relationships** are constituents of logical decomposition, conceptual and realization hierarchies, that describe a system as an interacting collection of objects
- ◆ **Properties and Functions** are inseparable from and definitive of elements, relationships, and the compositions that comprise them

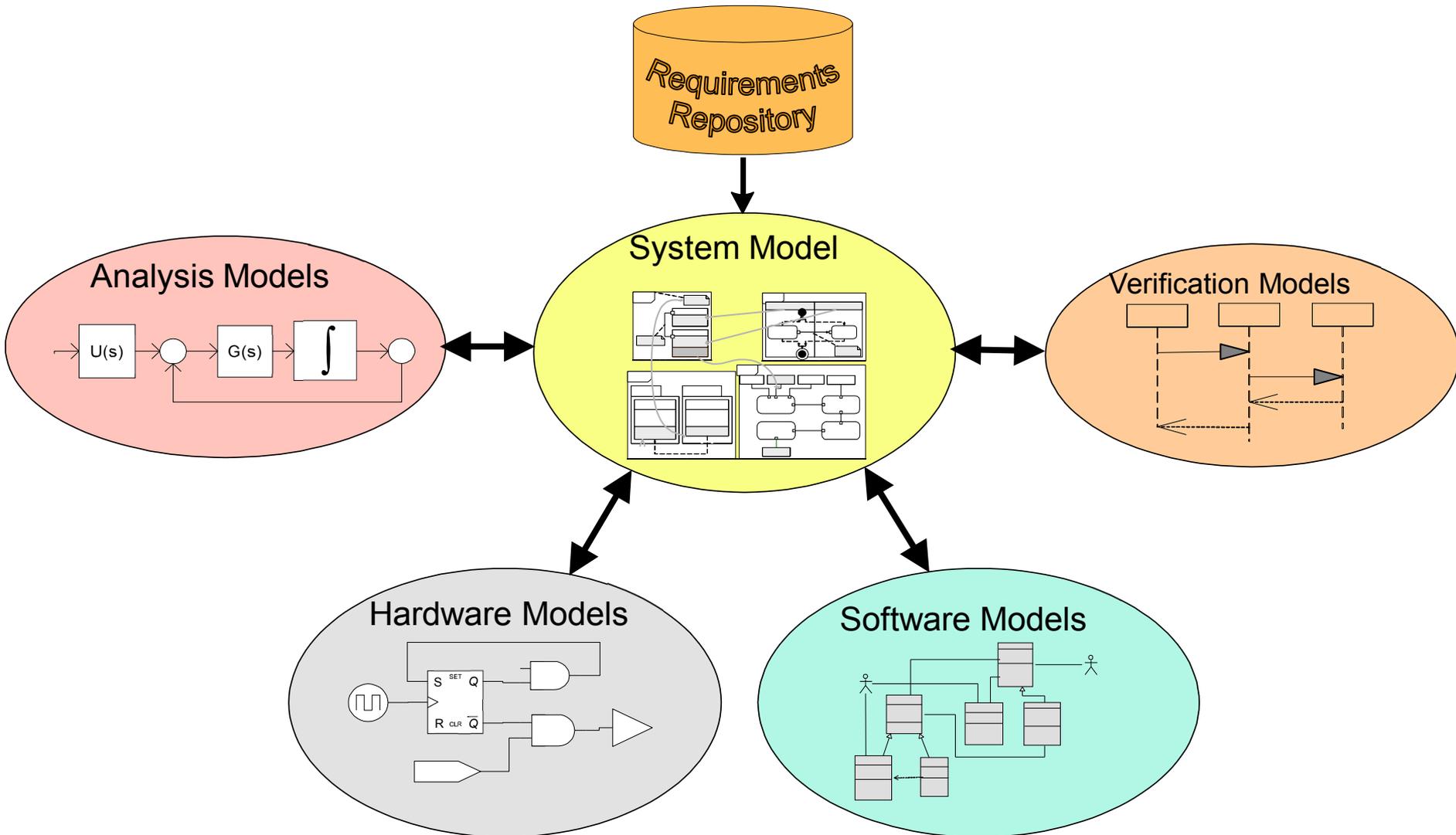
# Transitioning to Model-Based Methods

---



- ◆ In NASA's document-centric world, the dominant state systems engineering practice is the collection of information in stacks of viewgraphs accumulated (with supporting spreadsheets, graphics, etc.) in weakly organized file systems
- ◆ Systems engineers often devote a great deal of their time merely serving as the human ties among all of this data, too often at the expense of timely, creative, thoughtful, analytical efforts.
- ◆ Tools created explicitly with systems engineering in mind have made significant headway, many employing the Systems Modeling Language (SysML), an open standard defined in 2003 by INCOSE
- ◆ In regard to architecting, no firmly established pattern yet exists — let alone one fully suitable to NASA. It isn't apparent whether it is appropriate to even settle on a single framework that would cover our wide range of project types
- ◆ Model-based tools provide the environment and flexibility to define, explore, refine and utilize architecture frameworks that provide early value and can evolve to broad capability

# System Model as an Integrating Framework





# Path Forward

---

- ◆ **For architectures to be successful they need to be constructed with and for sustained stakeholder engagement, understanding and support**
- ◆ **We need development and adoption of an architecting framework (or frameworks), well integrated with current and evolving methodologies in the MBSE community**
- ◆ **Good architecting is more art than science and does not occur without strong support and commitment from the top. Pilot efforts alone are not sufficient. Support is needed from leadership at the center and agency levels to achieve critical mass to support project implementations effectively**
- ◆ **What ultimately matters is not architecting frameworks per se, but rather the elevation of architecting art and stature to a point where the purpose and value of a more structured approach is genuinely appreciated**

# Conclusions

---



- ◆ **Architecting and architectures can do much to resolve present deficits in systems engineering in general, and more importantly, for complex systems in particular**
- ◆ **Architecture, as a product in its own right, can act as a driving force for stable, integrated technical and programmatic development**
- ◆ **NASA is very good at engineering, but its programs, human and robotic, need sound, well crafted architectures to assure designs and implementation can and will meet stakeholder needs and achieve sustainable long term success**
- ◆ **An architecting framework, supported by model-based tools, can do much to simplify, clarify, and coordinate many aspects of NASA systems engineering practice and procedure, thereby improving the quality and performance of our missions, while helping to manage complexity within cost**