

# Applying an Agile Approach with MBSE

*THE VALUE OF PERFORMANCE.*

***NORTHROP GRUMMAN***



# Abstract

The combination of Model-Based Systems Engineering (MBSE) and the Agile process aids in the execution of programs that are constantly changing, are highly complex, and contain uncertainties throughout system development. Establishing an Agile MBSE approach allows teams to manage change and uncertainties, manage the complexity of a system, foster multidisciplinary collaboration between cross-functional teams, and provide a central repository model as the source of truth for system information.

Implementing an Agile MBSE approach provides teams the opportunity to easily manage changing system requirements and focus on Lean Modeling to deliver capabilities aligning to team-defined iteration goals. By breaking down model development into time-boxed iterations, the overall quality is increased allowing teams to focus on specific iteration goals to develop and collaborate upon. At the completion of each iteration, teams receive feedback to ensure customer alignment, overall focusing the team's efforts on Lean Modeling by avoiding Big Modeling Up Front (BMUF). This iterative process supports model development by adapting to change, ensuring transparency, delivering models frequently, and ensuring a short period of time where teams fail-fast to learn what specific approaches work. By defining the approach for combining Agile and MBSE best practices, the groundwork is laid for successful collaboration and generation of meaningful architectural solutions.

# Table of Contents

---

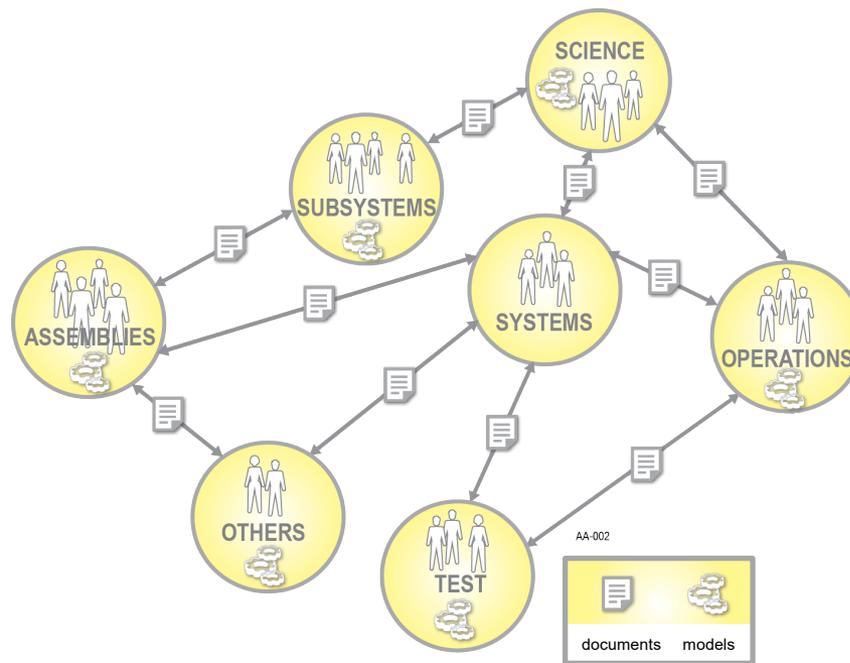
- Why Model-Based Systems Engineering (MBSE)?
- How do you apply MBSE in Agile Development?
- What are the challenges associated with Agile MBSE?
- Summary
- References

*THE VALUE OF PERFORMANCE.*  
***NORTHROP GRUMMAN***

# Why Model-Based Systems Engineering (MBSE)?

# Key Systems Engineering (SE) Challenges

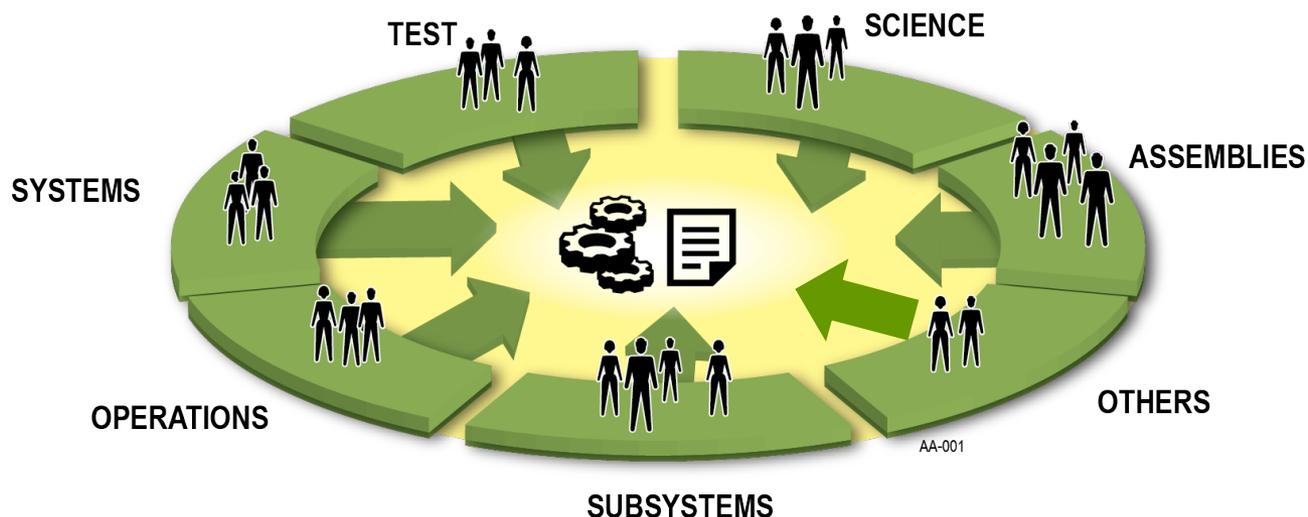
- Customers do not have detailed/justified definitions of their problems
- The **complexity** of future **problems** and **solutions** (e.g., hardware, software, systems, systems-of-systems) is **growing exponentially** and will (or have) reach a point where we can no longer afford to solve them
- Using documentation to **manually** manage SE data, information and knowledge is exceptionally labor-intensive and error-prone for complex systems (of systems)
- Producing and using voluminous documentation to share and transfer SE knowledge is ineffective
- Assessing the impact of changes is exceptionally difficult in complex solutions



**Government programs involving complex solutions must address some enduring SE challenges**

# Model-Based Systems Engineering (MBSE)

- MBSE is a standardized environment/methodology that provides linkage, rules, metrics and views of engineering artifacts created to define, solve, and manage **problems and solutions of high complexity**
- MBSE models are the central repository of all engineering/management artifacts—**single source of truth**
- MBSE modeling tools are used to build and evolve models, described using modeling languages



**Government programs increasingly employ MBSE throughout the solution lifecycle**

# Models

- **What is a Model?**

- A simplified—physical, mathematical and/or logical—representation of a mission/solution at some particular point in time or space intended to promote understanding of the real mission/solution
- Provides insight about one or more of the mission/solution's aspects, such as its function, structure, properties, performance, behavior, or cost
- Leveraged by most engineering disciplines (e.g., electrical, mechanical, software)

- **Types of Models**

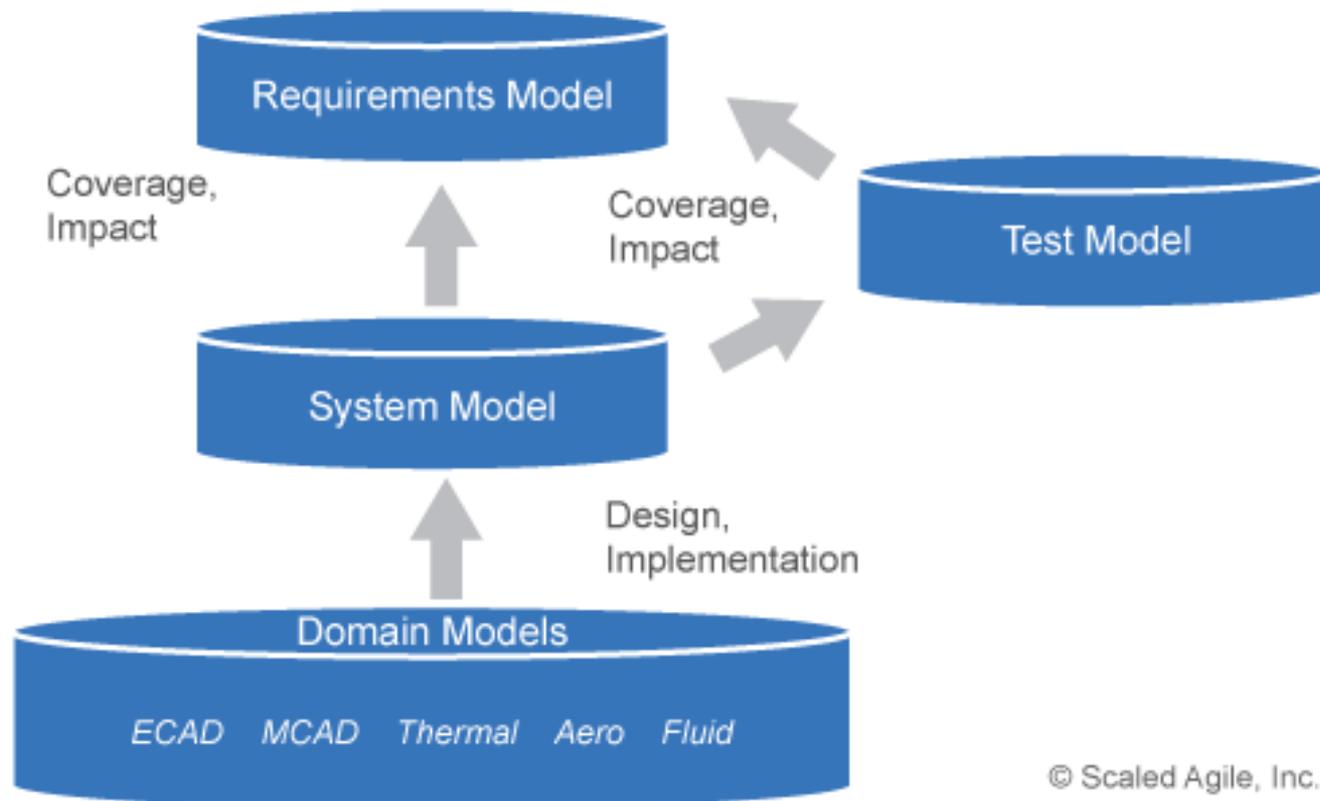
- [Descriptive](#)
- [Analytical](#)
- [Hybrid](#) (Descriptive + Analytical)
- [Domain-Specific](#)
- [Mission/Solution/System](#)
- [Integrated/Executable](#)

- **Simulations**

- [Implement models over time](#)
- Used to analyze the complex dynamic behavior of missions, solutions and their respective operational environments
- May be [live, virtual or constructive](#)

**“All models are wrong, but some are useful.”—George E. P. Box**

# Examples of Models



**Government programs will develop numerous models during the period of performance**

## Uses of Models

- Characterizing an existing mission/solution
- Mission/solution concept formulation, exploration and evaluation
- Solution design synthesis and requirements flow-down
- Support for solution integration and verification
- Support for training solution operators, maintainers and other stakeholders
- Capture and sharing of solution knowledge
- Support change impact analysis, as part of on-going solution design evolution
- Estimate mission/solution costs
- Evaluate mission/solution alternatives
- Efficiently produce contract- or compliance-required documentation

**Government programs use models to help answer critical SE questions**

# Benefits of MBSE

- Single source of engineering truth
- Increase ability to manage complexity
- Improve communications among stakeholders
- Rapidly analyze the impact of changes
- Improve system architecture, design and quality
- Minimize staff attrition impact
- Enable fast-feedback on requirements and design decisions
- Lower maintenance costs to modify solution design
- Validate problem and solution hypotheses at lower cost
- Identify risks earlier
- Improve productivity through automation and reusability

**Government programs employing MBSE realize numerous benefits—at a cost**

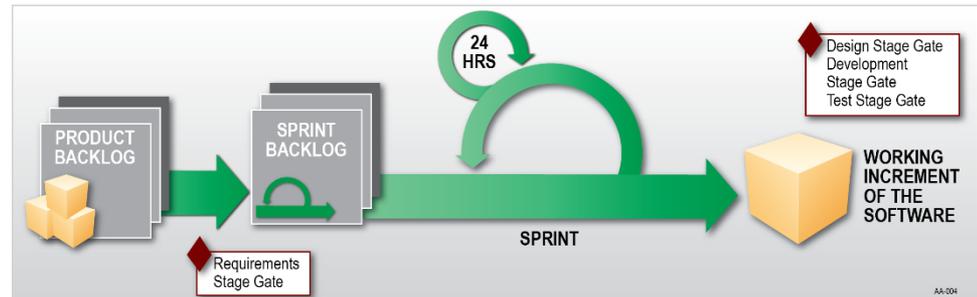
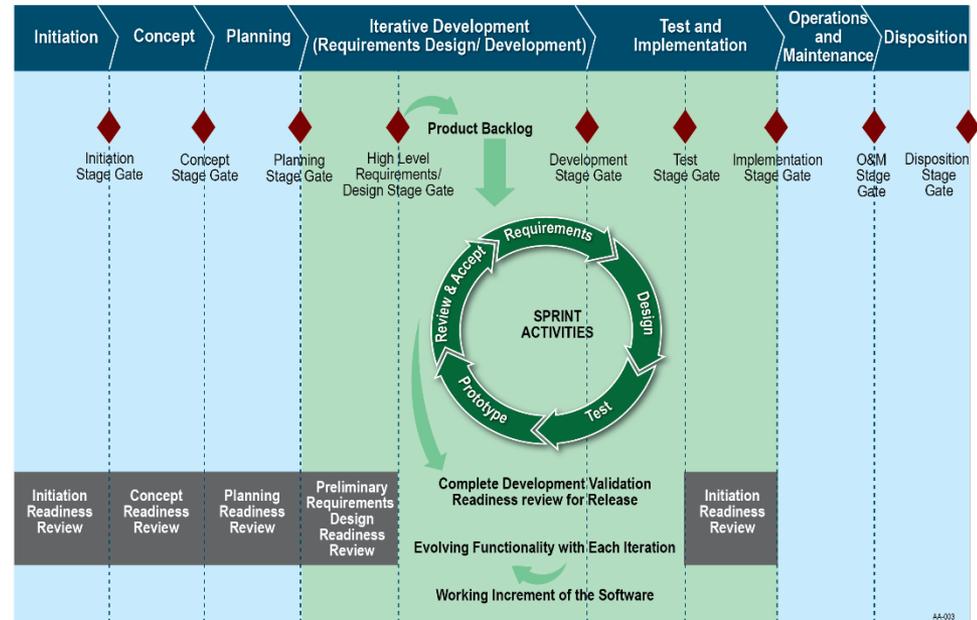
*THE VALUE OF PERFORMANCE.*  
***NORTHROP GRUMMAN***

# How do you apply MBSE in Agile Development?



# Agile Development

- Traditional Agile development frameworks and best practices (e.g., [Scrum](#), [XP](#)) don't explicitly provide guidance on how to:
  - Integrate **MBSE** and other SE disciplines
  - Scale to large, distributed project teams (potentially with many suppliers) developing **complex solutions**
  - Manage projects in heavily regulated domains such as health, financial, **government**, etc.
  - Operate, maintain and evolve systems in production

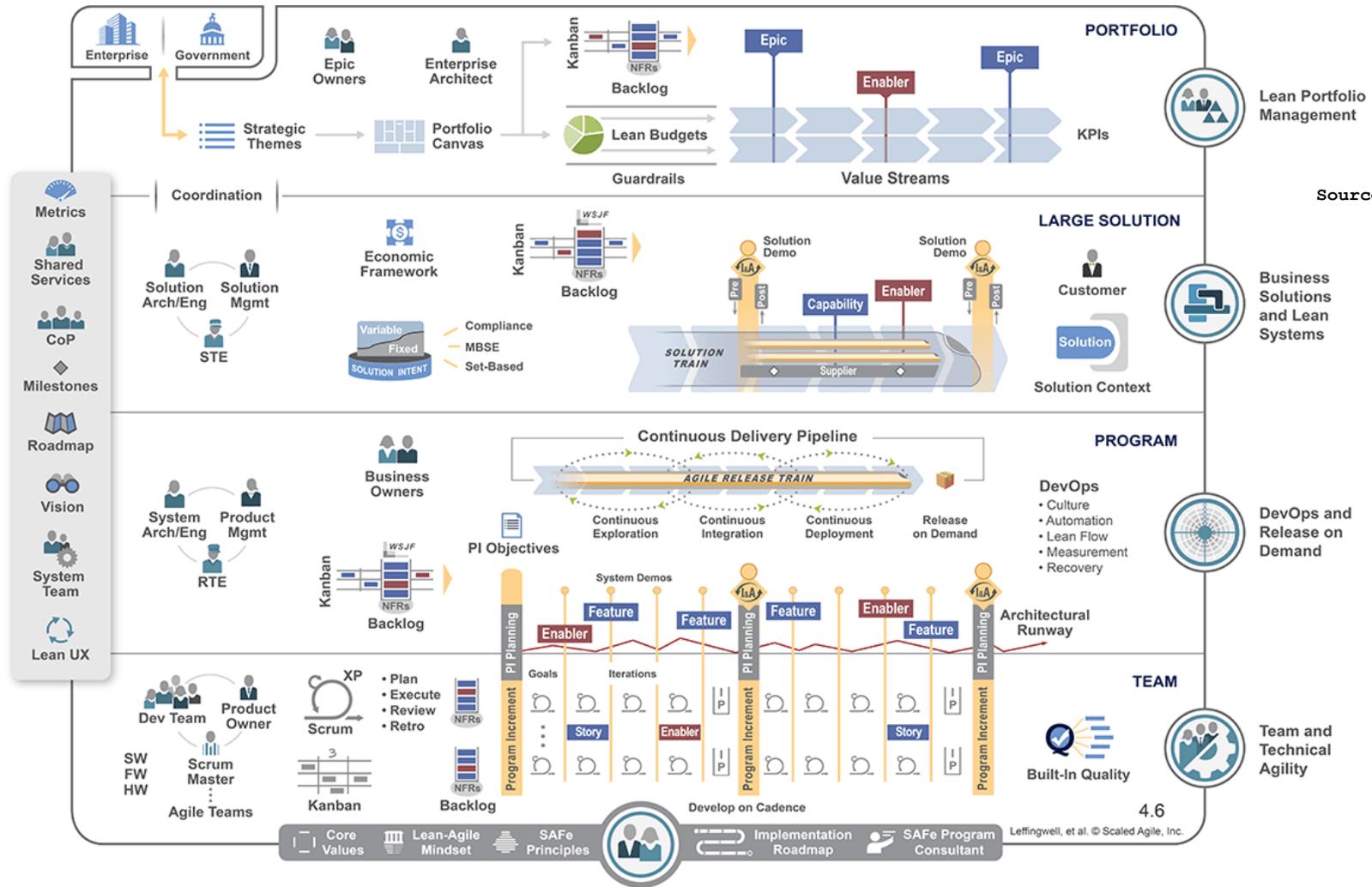


*"We embrace modeling, but not in order to file some diagram in a dusty corporate repository."*—Jim Highsmith, [History: The Agile Manifesto](#)

**Defense Innovation Board Issues Guide to 'Detecting Agile BS'**

**Most government programs that develop any software are expected to employ Agile**

# Agile Development at Scale

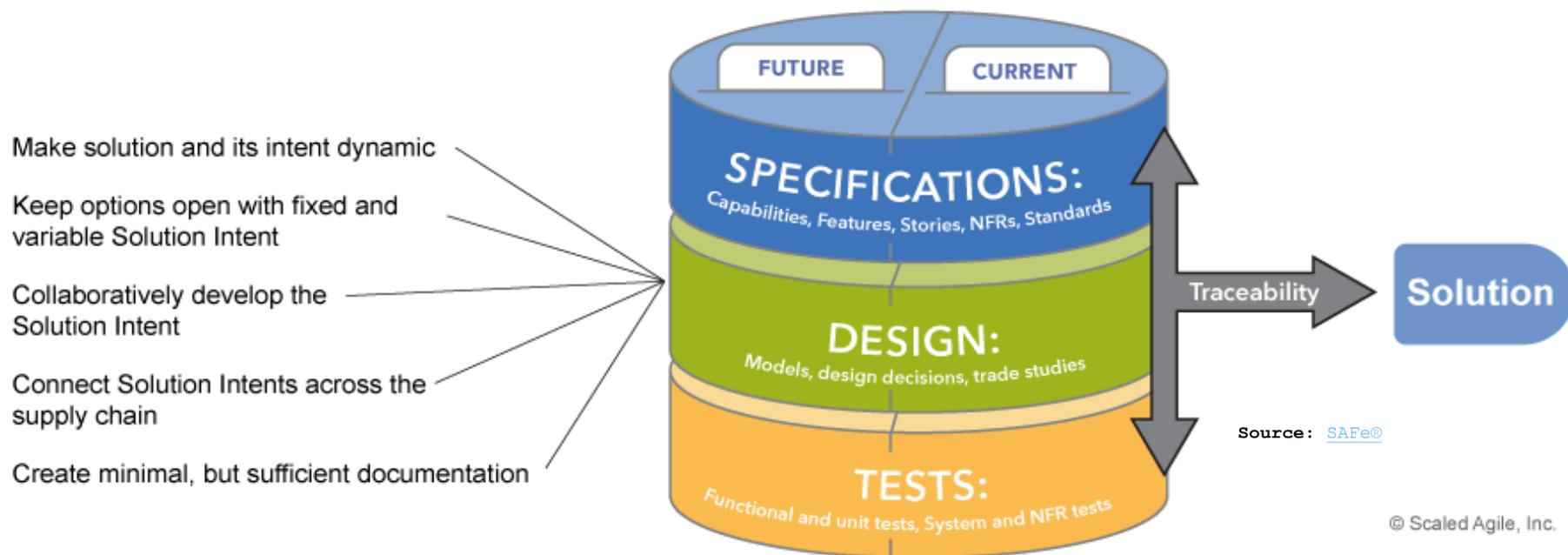


Source: [SAFe®](#)

**Government agencies are starting to embrace the Scaled Agile Framework (SAFe®), which incorporates MBSE**

# Agile MBSE (AMBSE) Best Practices

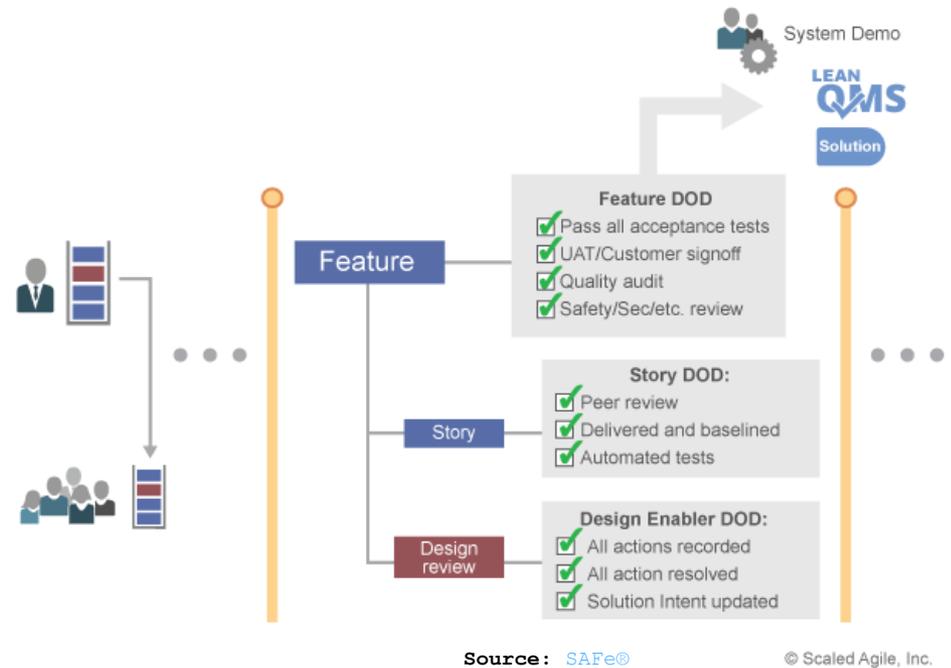
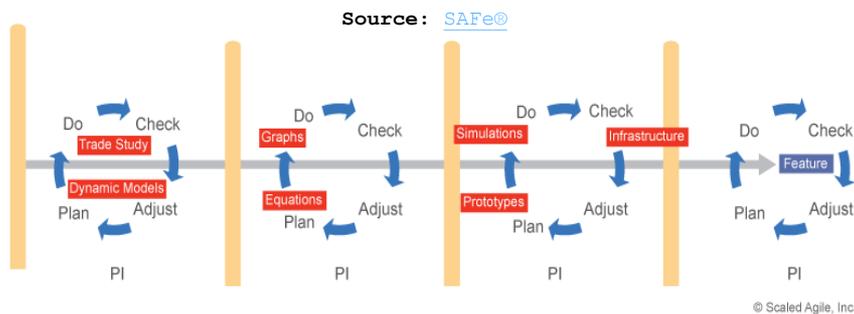
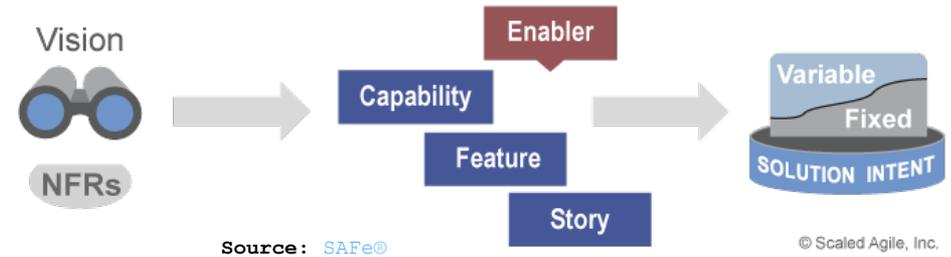
- Incrementally build “just barely good enough” models of the Solution Intent (“as-built” and “to-be”) with each Iteration
  - Avoid Big Modeling Up Front (BMUF)



**Apply AMBSE when there is significant complexity, uncertainty and/or ambiguity during solution development**

# Agile MBSE (AMBSE) Best Practices

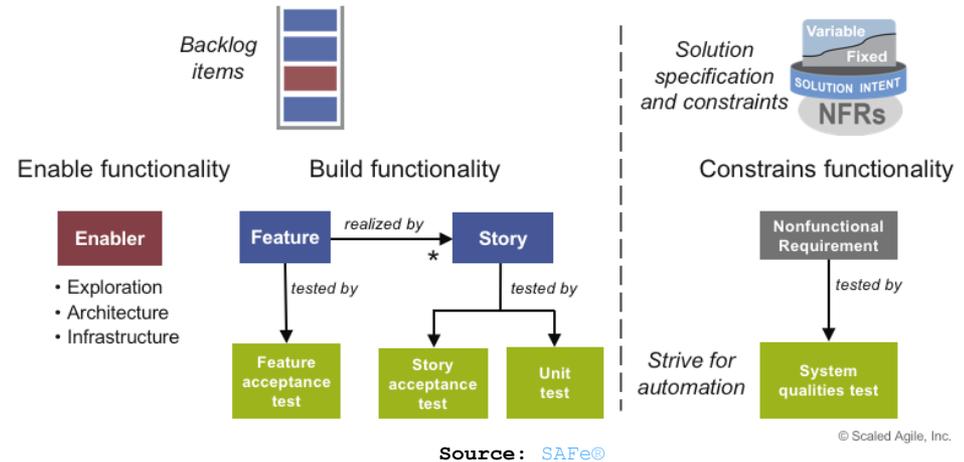
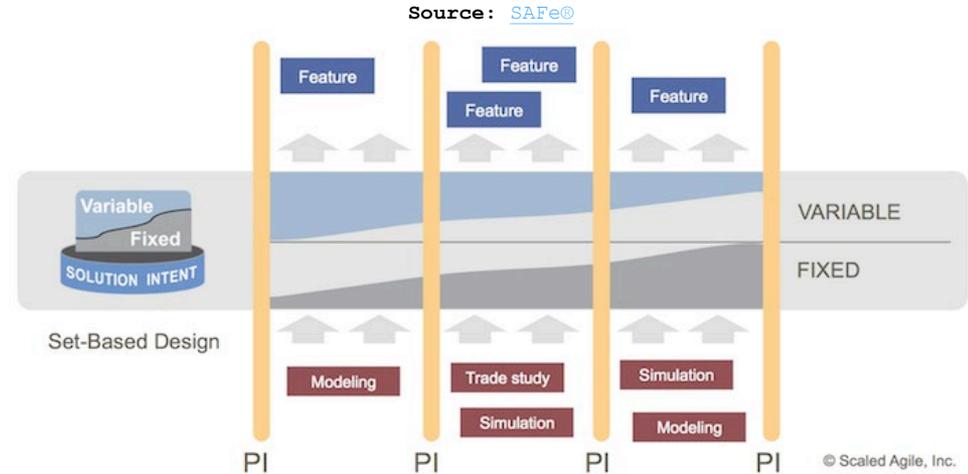
- Describe modeling-related Acceptance Criteria and/or Definition of Done on the Solution's Capabilities and Features
- Describe explicit modeling work as Enablers associated with the Solution's Capabilities, Features and/or Non-Functional Requirements



**Apply AMBSE when there is significant complexity, uncertainty and/or ambiguity during solution development**

# Agile MBSE (AMBSE) Best Practices

- Utilize models to support [Set-Based Design](#) and [Compliance](#) (e.g., V&V, RMF) activities
- Ensure models are [configuration controlled](#)
- Employ [role-based access control](#) to models
- Create testable and executable models (when feasible)
  - Utilize [Behavior-Driven Development](#) (BDD)
  - Utilize/develop model verification scripts



**Apply AMBSE when there is significant complexity, uncertainty and/or ambiguity during solution development**

# Agile MBSE (AMBSE) Best Practices

- [Maximize stakeholder participation](#) in identifying, building and validating models
- Create and capture informal models developed as part of refining the [Solution Backlogs, Program Backlogs](#) and [Team Backlogs](#) (i.e., “[model storming](#)”)
- [Minimize duplication of model elements](#)
- Model at the lowest security classification possible
- Establish modeling standards (e.g., processes, style guide) across all projects in the [Portfolio](#)
- Leverage [Iteration Retrospectives](#) and [Inspect & Adapt](#) workshops to improve modeling standards

**Apply AMBSE when there is significant complexity, uncertainty and/or ambiguity during solution development**

# Agile MBSE (AMBSE) Best Practices

- Create models working in pairs
- Conduct a modeling “jam session” after [Iteration Planning](#) with modelers across the [Agile Release Train](#)
  - Review models created in the previous [Iteration](#)
  - Share lessons learned & best practices
  - Collaborate on the most difficult modeling problems in the current [Iteration](#)



**Apply AMBSE when there is significant complexity, uncertainty and/or ambiguity during solution development**

*THE VALUE OF PERFORMANCE.*  
***NORTHROP GRUMMAN***

# What are the challenges associated with Agile MBSE?

# Agile MBSE (AMBSE) Challenges

- Culture change
- Leadership buy-in
- Building and retaining modeling talent
- Access to and availability of stakeholders to support iterative model validation
- Adopting emerging modeling standards (e.g., UAF 1.0, SysML 1.5)
- Lack of training in modeling methodologies, languages, frameworks and tools
- Vendor lock-in with modeling tools
- Communicating models with stakeholders with no experience with modeling languages and notation

**Government programs that embrace AMBSE will still face a number of challenges**

# Agile MBSE (AMBSE) Challenges

- Integration of multiple modeling tools and associated data, as well as simulators, within and across security classifications
- Integration of modeling tools with agile project management and software development tools
- Tools, modeling notations, and processes used by one discipline (e.g., Systems, Hardware, or Software) do not mesh well with other disciplines
- Semantic gaps between business/mission and engineering domain & disciplines
- Creating executable models
- Automated document production snafus
- Use of models in [DevSecOps](#) (at scale) and [Software Factories](#)

**Government programs that embrace AMBSE will still face a number of challenges**

***THE VALUE OF PERFORMANCE.***

***NORTHROP GRUMMAN***

