Raytheon

Customer Success Is Our Mission

Informal Symposium on
Model-Based Systems Engineering
DSTO, Edinburgh, South Australia

Does a Model Based Systems Engineering Approach Provide Real Program Savings? – Lessons Learnt

Presenter: Steve Saunders
FIEAust CPEng

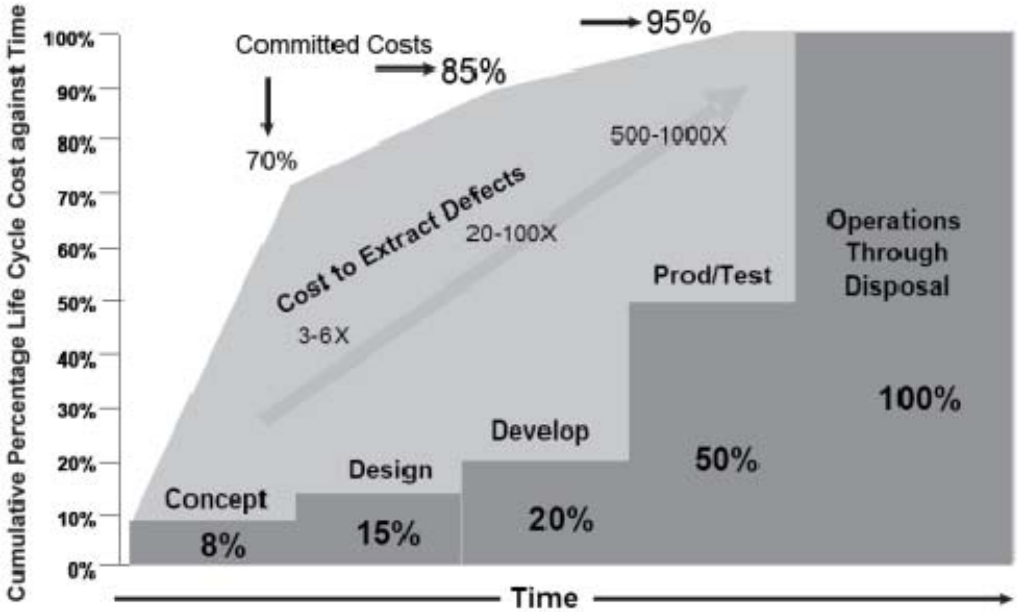
AWD Combat System Chief Engineer
Date: 25 Oct 2011

Agenda

- Background
- Document Centric Vs Model Based Systems Engineering
- Coping With Complexity
- Deploying MBSE on a Program – Lessons Learnt
- Extending Model Based Analysis Back to Architecture Definition
- Conclusions
- Questions
- Glossary / References

Background

- One recognised source of Program Failures is the result of poor Requirements Definition
- Total Program Cost is Committed in the early Phases of Programs



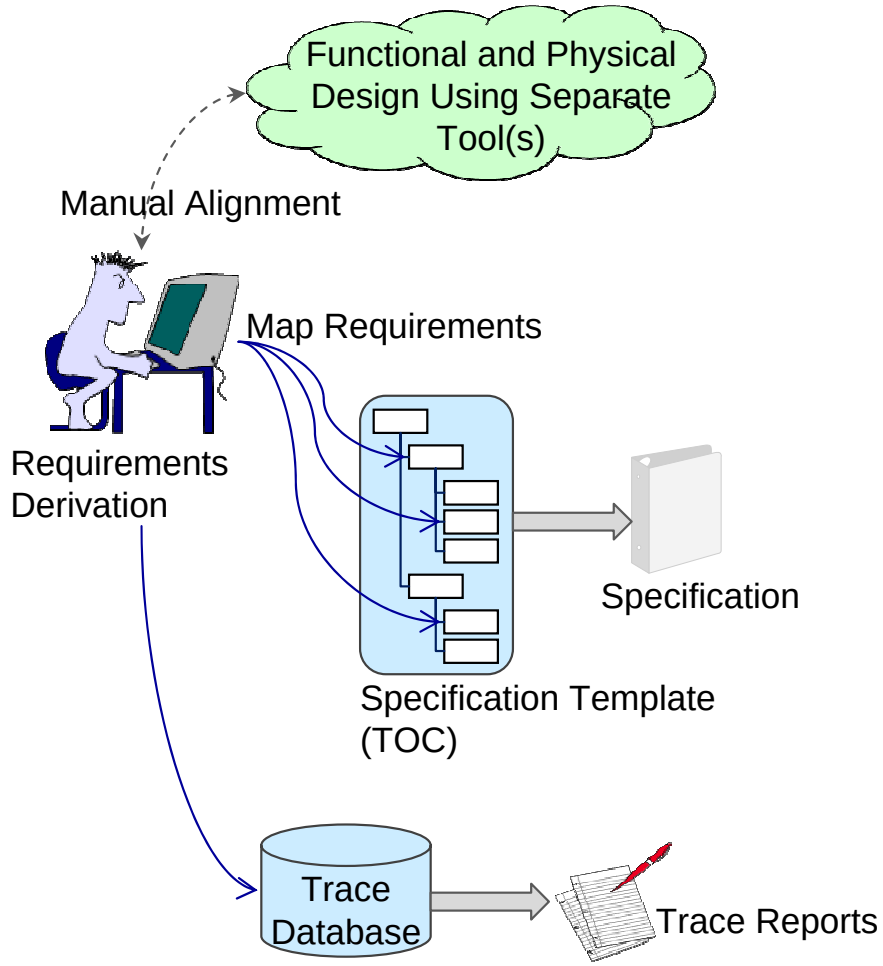
Defense Acquisition University, 1993

- Hence, mechanisms to remove requirements errors up front should Mitigate Program risk
- Postulate MBSE helps achieve this…

Raytheon

Document Centric Vs MBSE

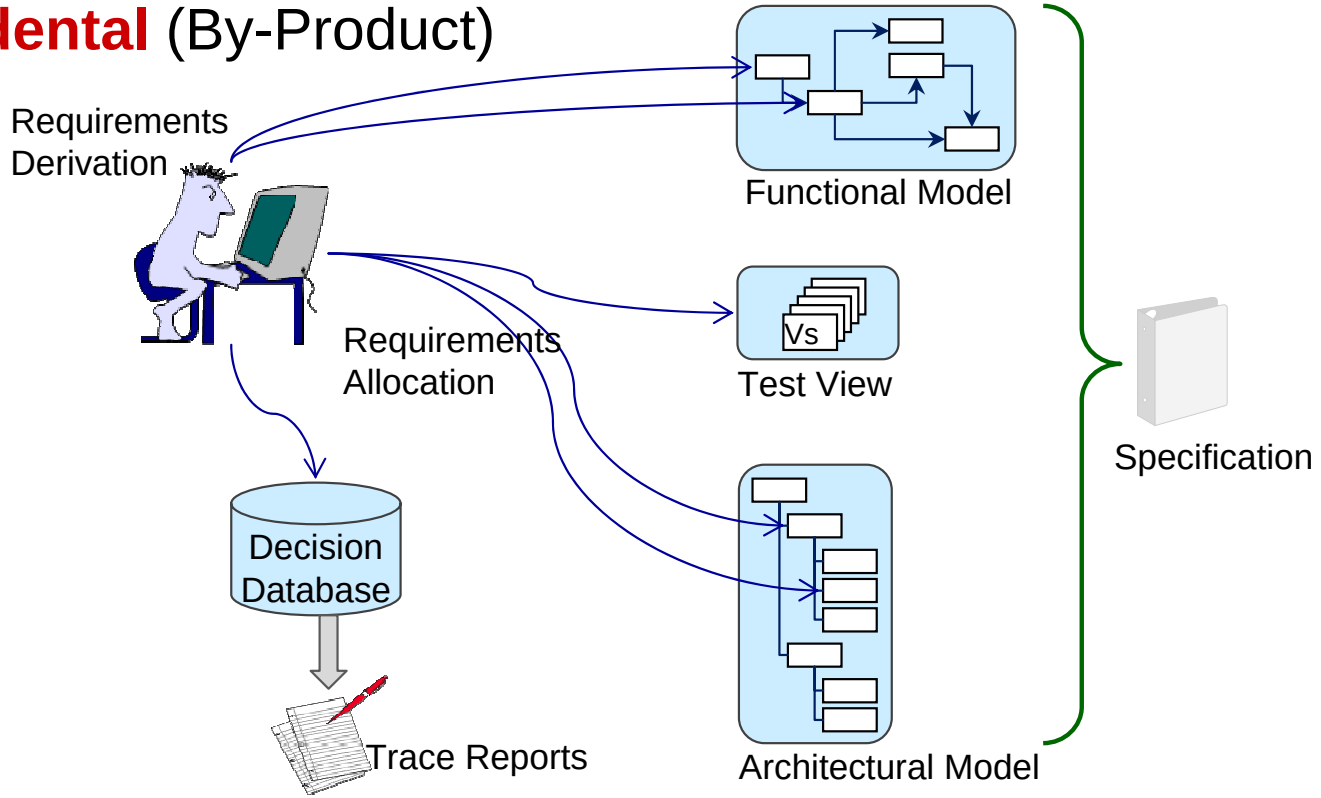
- Document Centric Systems Engineering
  - Focus is the Specification
  - Tool of Choice
    - Requirements Management Tools
  - Quality
    - Traceability Reports?
  - Measure of Completeness
    - Page Count?
  - Correctness
    - Specification Structure is a Static Functional Model (Problem)



Requirements Specification is the Focus

Document Centric Vs MBSE

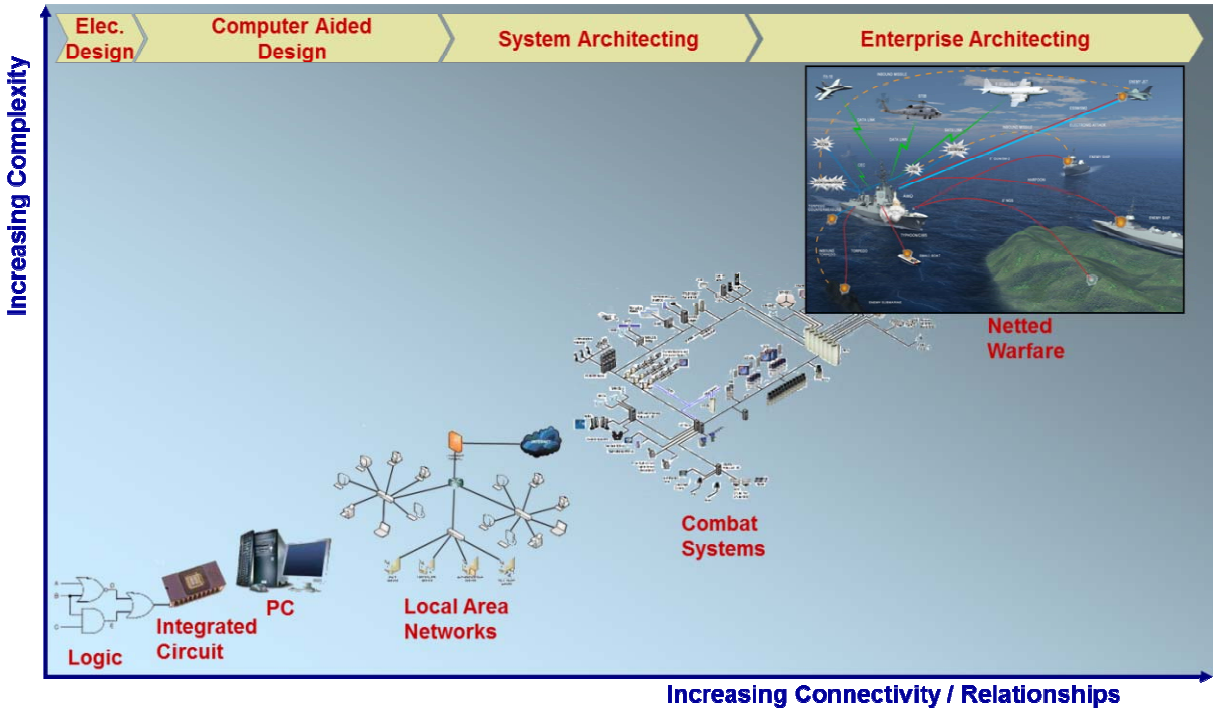
- Model Based Systems Engineering (MBSE)
  - Understanding the System Behaviour
  - Relating Requirements to Functions
  - Complete all Views of the Model
  - Specification is **incidental** (By-Product)



Requirements Specification is a By-Product

Coping With Complexity

- All but trivial systems involve Complexity beyond the ability of the human mind to comprehend in a complete viewpoint *Miller[1956]*
- Behaviour of System needs to be Understood before requirements can be derived



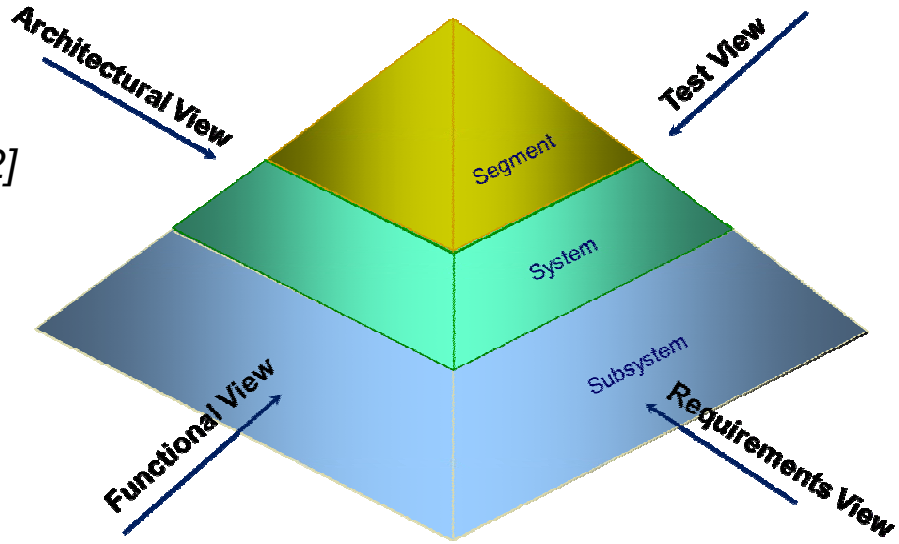
- Why do some Practitioners believe a Requirements Specification can be Written in Isolation to a behaviour model?

Deploying MBSE on a Program – Lessons Learnt

Raytheon

- Background
  - Implement MBSE using Systems Engineering Tool
  - Focus on First Principles
  - Pre-Dated more formal MBSE approaches

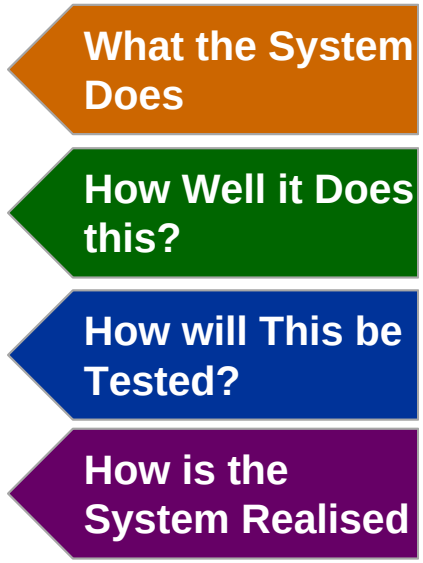
- Approach
  - Minimum of Four Views defined *Mar[2002]*
    - Functional View
    - Requirements View
    - Architectural View
    - Test View
  - All views linked and related
  - All views developed / refined as the model is matured



Deploying MBSE on a Program – Lessons Learnt

Raytheon

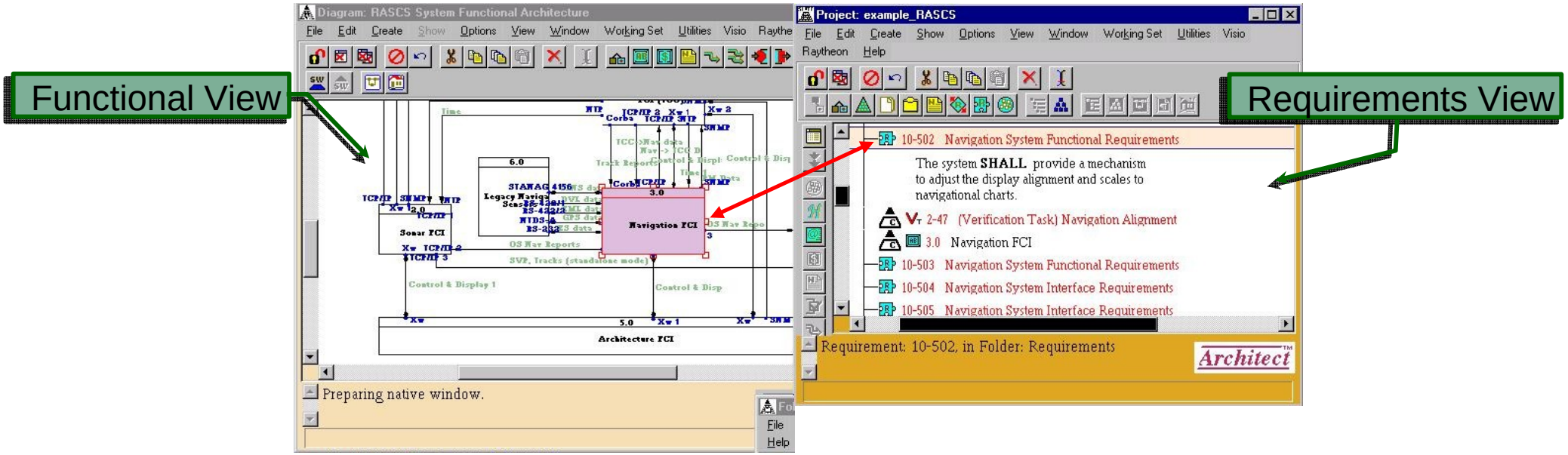
- Systems Engineering Tool Tailoring
  - System Level Automation Tools for Engineers (SLATE™) Employed
    - System functional behaviour modeled
    - Functional Hierarchy generated from the model
    - Requirements developed for the functional elements
    - Verification Statements captured for every requirement developed
    - Abstraction blocks/hierarchy used to model the System Breakdown Structure
    - Specification was not the Primary Work Product – Generated from the model as an Artifact



Keep the Model Simple, Understand the System Behaviour, Develop the Model, Don't Focus on the Specification – ALLOW THE TOOL TO GENERATE THE SPEC

Deploying MBSE on a Program – Lessons Learnt

Raytheon

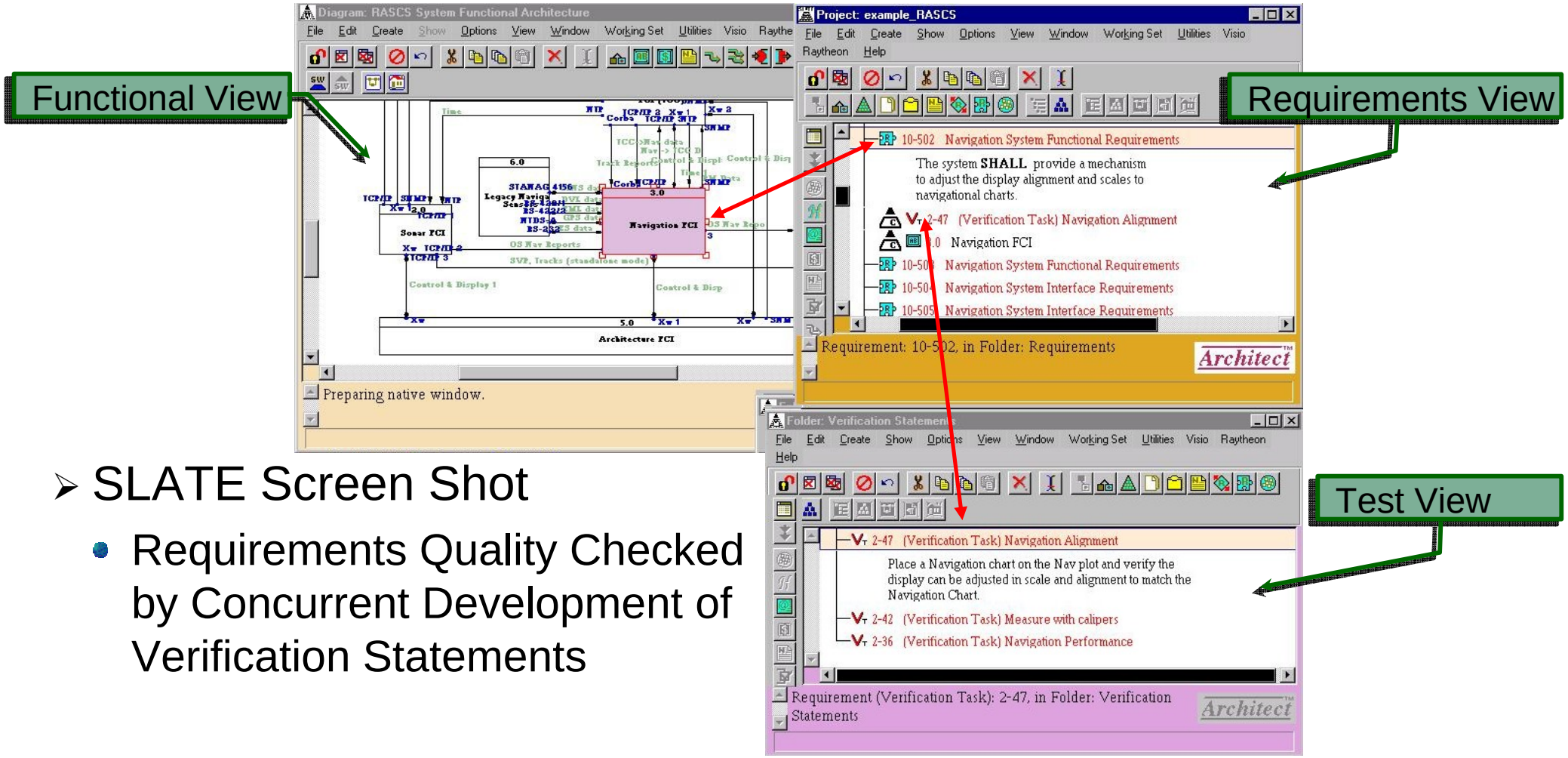


- SLATE Screen Shot
  - Functional Model and Requirements Developed Concurrently
  - **WHAT** the system does Developed alongside **HOW WELL** the System does this

Functional Model and Functional Requirements developed Concurrently

Deploying MBSE on a Program – Lessons Learnt

Raytheon

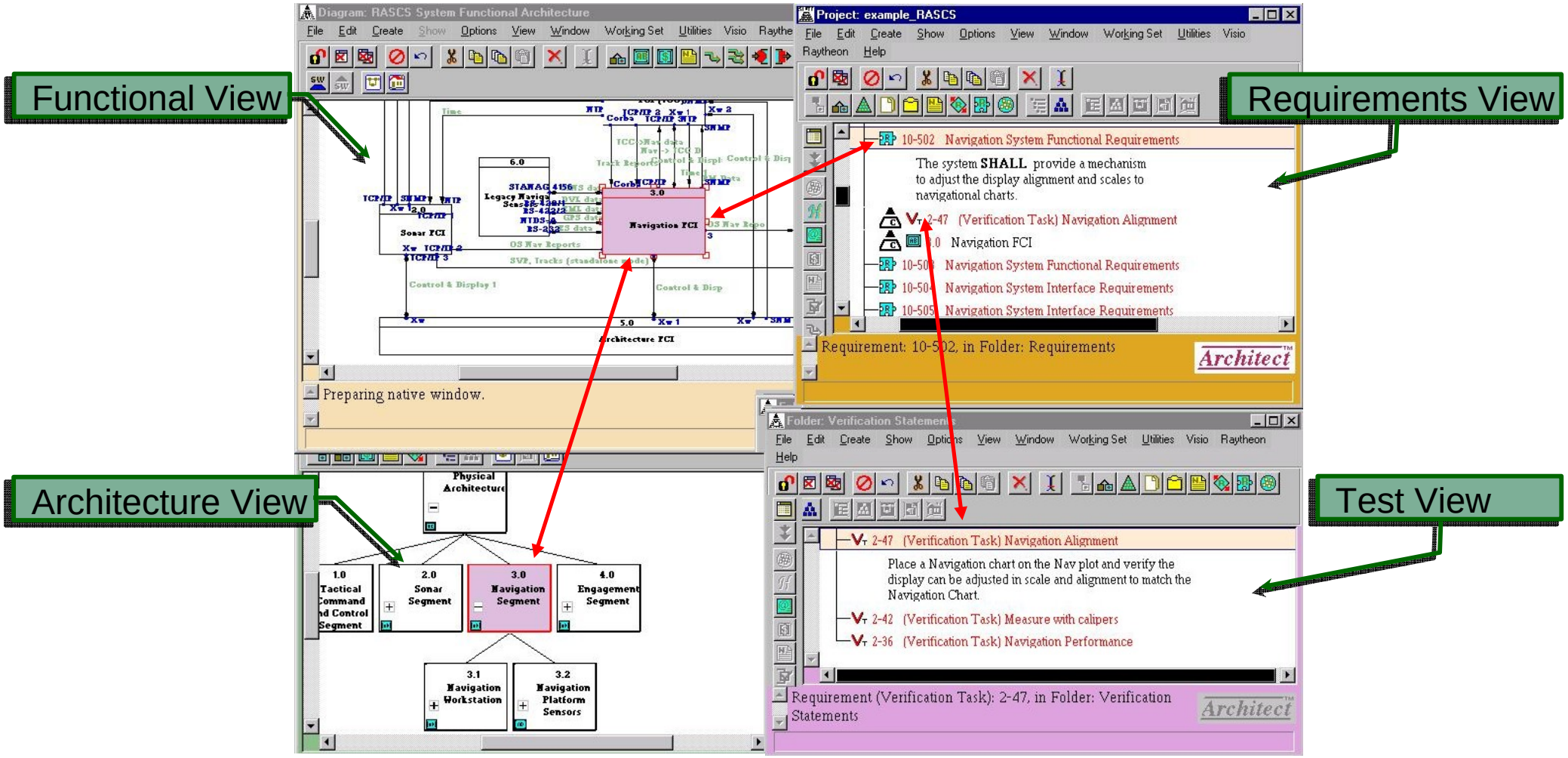


- SLATE Screen Shot
  - Requirements Quality Checked by Concurrent Development of Verification Statements

Test Requirements by Developing Verification Statements with Requirements

Deploying MBSE on a Program – Lessons Learnt

Raytheon



Synthesise Design – Link Functions to Physical Elements

Informal Symposium on
Model-Based Systems Engineering
DSTO, Edinburgh, South Australia

Deploying MBSE on a Program – Lessons Learnt

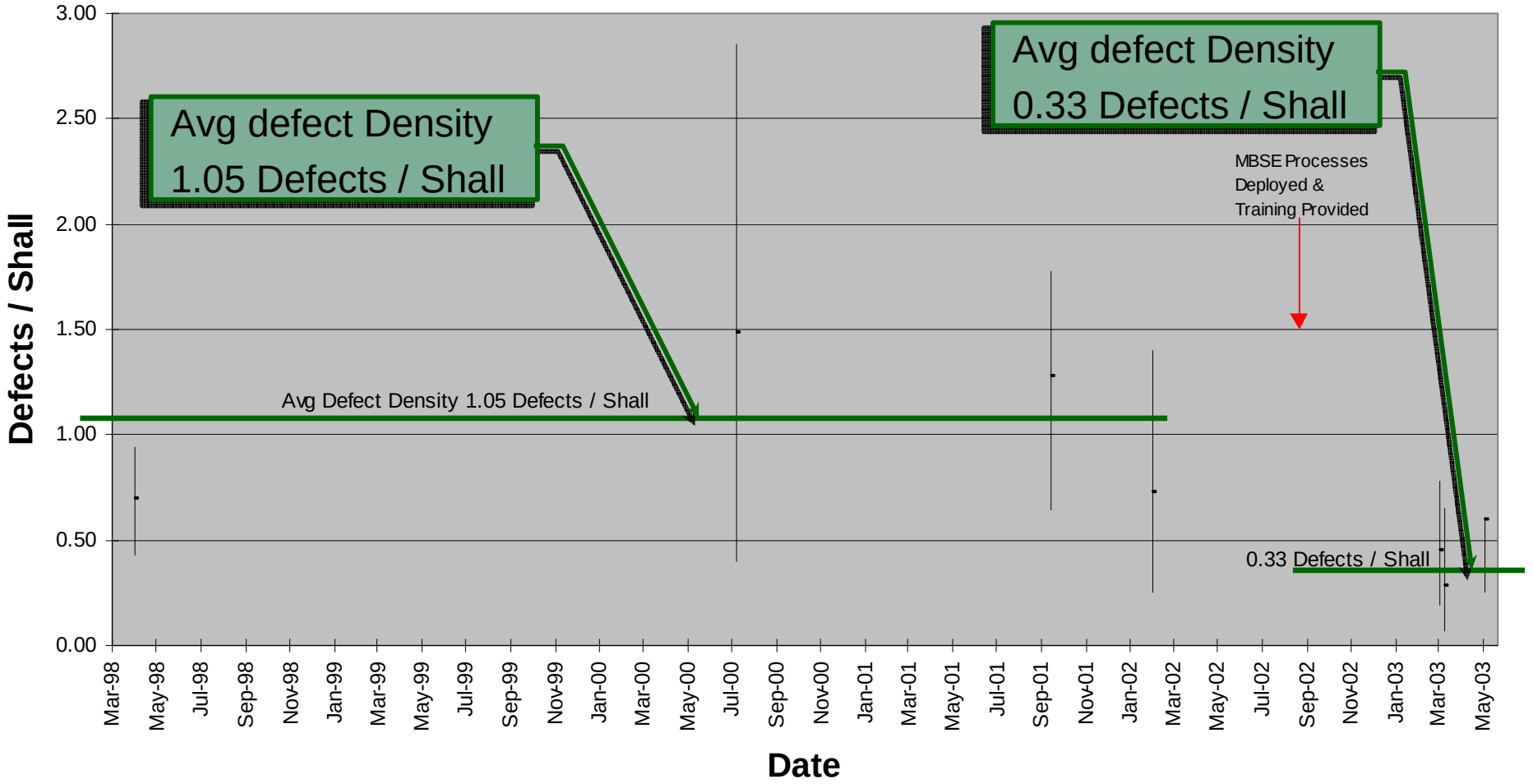
Raytheon

- Question: Is there a Benefit in using MBSE on Programs?
  - Assuming one direct Program outcome is related to specification errors
    - Define a Criteria for Measuring Specification Defects *Gilb[2002]*
    - Sample a set of specifications to quantify specification defects
    - Use an Independent Review Team for sample review
    - Specifications over a 5 year period were reviewed
      - Covered 4 “Traditional” Requirements Definition Programs
      - Covered 3 Programs using MBSE approach
- Results of Sample Audit Provided Next…

Deploying MBSE on a Program – Lessons Learnt

Raytheon

Specification Defects (Per Shall)



68% Reduction in Specification Defects since MBSE Practices Introduced

Extending Model Based Analysis back to Architecture Definition

Raytheon

- Question: Where is the Value in Systems Engineering?

  - To a Systems Engineer –
    - "Brings multi-disciplinary engineering processes"
    - "Follows a proven process"
    - "Systematic decomposition of complexity to allow system elements to be built"
    - etc.

  - To the Business or Enterprise Stakeholders –
    - "Must demonstrate how the system meets the enterprise needs"
    - "Must show how current problems are solved for the business"
    - "Must meet the stakeholder expectations"
    - "Provides a Return on Investment"
    - etc.

Extending Model Based Analysis back to Architecture Definition

Raytheon

- Good Systems Engineering, or, Employment of MBSE helps develop the System Right
  - Captures functional behaviour – what the system does
  - Helps ensure requirements are coherent with what the system does
  - Helps ensure consistency of terms

MBSE Helps Develop the System Right

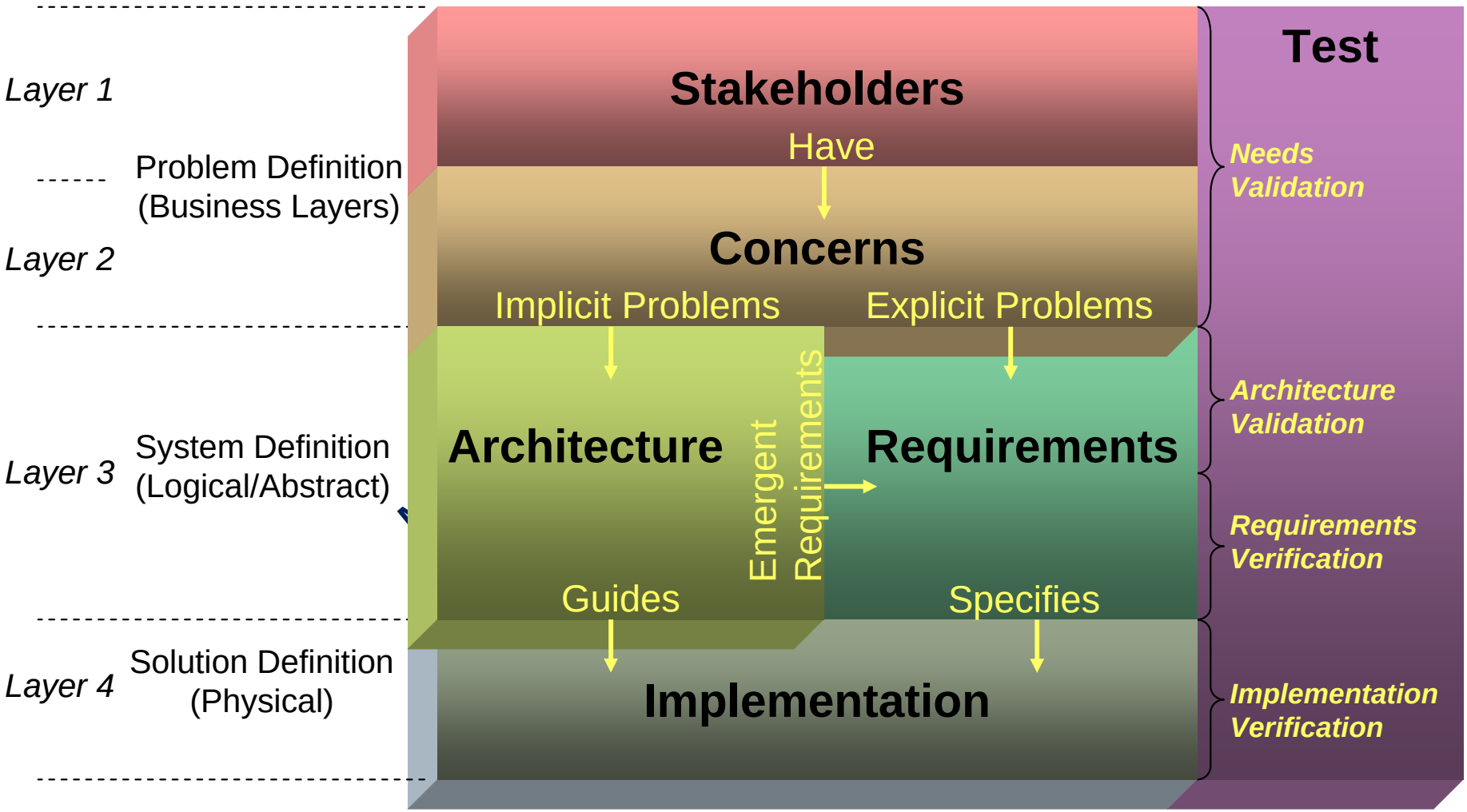
- Does it ensure the Right System is Defined?
  - System/Enterprise Architecting (in part) looks at the Business Needs / Values
    - Helps scope and align the System to meet the Business/Enterprise Needs
    - Helps align the System to Transition from Current to Future Architecture

System Architecting Helps Define the Right System (by defining the Problem)

System Architecting uses Models – why not Link with MBSE Models?

Extending Model Based Analysis back to Architecture Definition

Raytheon



[Saunders2007]

System Architecting Helps Define the PROBLEM. MBSE Continues Process Through to Design

Conclusions

- Programs are sensitive to errors during Requirements Definition
- Requirements Definition should first consider what the System does (its Functional Behaviour)
- System Functional Behaviour cannot be expected to be understood to the extent needed to create a complete/consistent Specification
  - System Functional Modeling must be undertaken
  - Functional Modeling should be linked to the efforts in defining requirements
- Adoptions of elementary MBSE has demonstrated significant reductions in requirements errors
  - Similar results expected from more formal methods (SysML)
- MBSE should not be constrained to commence with Requirements; Propose the model should link into Architectural Modeling

Raytheon

Questions

Steve Saunders FIEAust CPEng
AWD Combat System Chief Engineer

steve.saunders@ausawd.com
+612 8870 6648

Raytheon

Glossary / References

Glossary

- MBSE — Model Based Systems Engineering
- SLATE — System Level Automation Tools for Engineers

References

Gilb, Tom, *Requirements-Driven Management,* Draft book manuscript found at http://www.result-planning.com, 5 Sept 2002

Mar, Brian W. and Morais, Bernard G., *FRAT - A Basic Framework for Systems Engineering,* Proceedings of the INCOSE 2002 Symposium.

Miller, George, *"The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information",* The Psychological Review, 1956, vol. 63, pp. 81-97

Saunders, Steven *"Promoting the Real Value of Systems Engineering using an Extended SCARIT Process model"*, Proceedings of the INCOSE 2007 Symposium, 2007

.

Author

- **Steve Saunders** is a Raytheon Certified Architect and an engineering fellow for Raytheon Australia. He received his bachelor of electrical engineering, from the University of Technology Sydney (UTS) with first class honors in 1990. He has worked with Rockwell International, Boeing Australia and now Raytheon Australia on major Australian defense projects in various systems engineering management, requirements development, architecture, design and test roles. Steve was the engineering manager and design authority for the Collins Replacement Combat System and is currently the chief architect on the Royal Australian Navy's SEA 4000 Air Warfare Destroyer program. Steve has written numerous articles on systems engineering and enterprise architecting and has a strong interest in improving system engineering maturity and the agility of systems engineering to support the rapidly evolving technology environment and complexity within the defense industry.