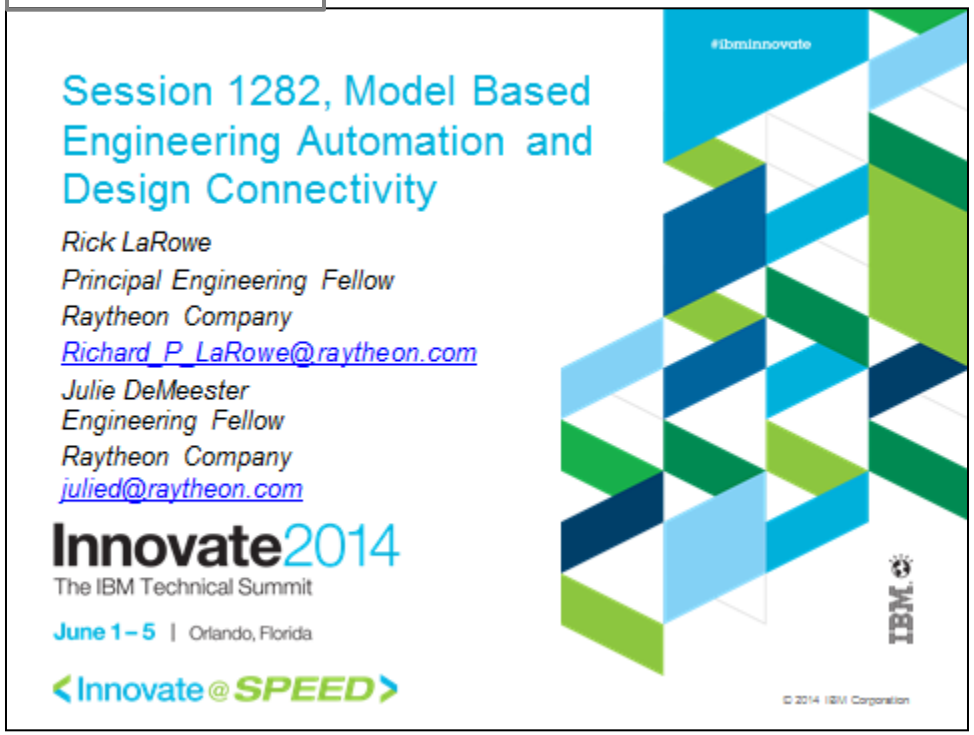
INCOSE IW 2015
MBSE Workshop

- **Session Title:**
  - **Open Services for Lifecycle Collaboration (OSLC)**
- **Talk Title:**
  - **Raytheon's Experience with OSLC**
- **Speaker:**
  - **Ron Williamson**
    - **Raytheon**
    - **Senior Engineering Fellow**
    - **ron_c_williamson@raytheon.com**

Taken from

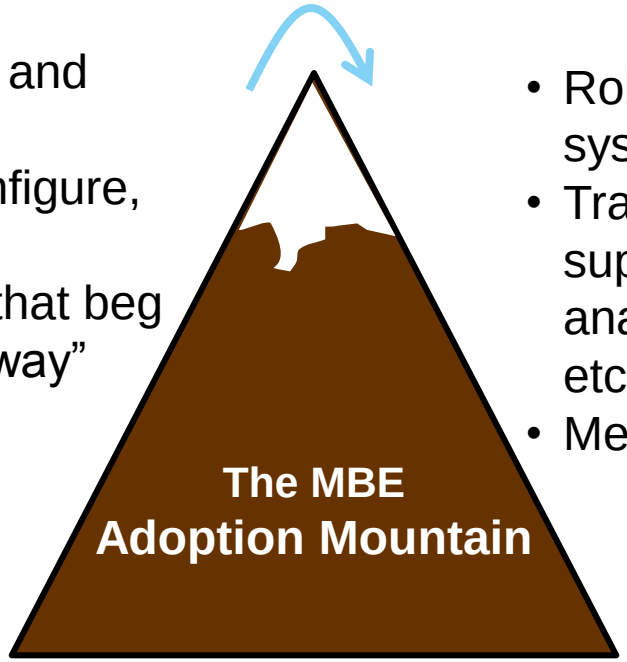


Motivation

- Adopting Model Based Engineering can be challenging for an organization
- Requires **INVESTMENT** with unproven **PAYBACK**
- Demands an approach to “get over the adoption mountain” quickly and effectively

- New processes to define and learn
- New tools to acquire, configure, deploy, and learn
- Legacy artifacts in hand that beg to be leveraged “the old way”
- MBE needs to be proven

INVESTMENT

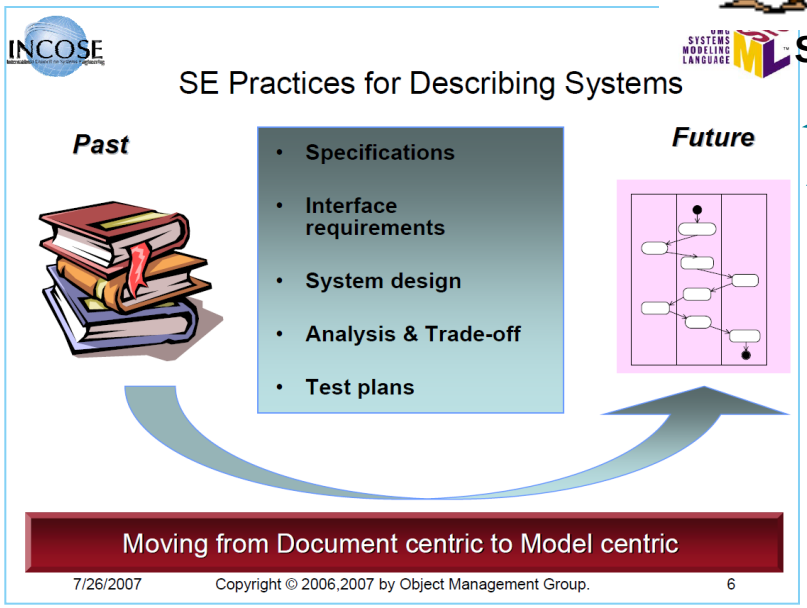


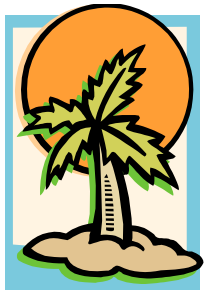
- Robust models that drive the system engineering process
- Traceability and linking to support change impact analyses, system evolution, etc.
- Metrics and assessments

PAYBACK

Challenges – Integrating Islands of Models

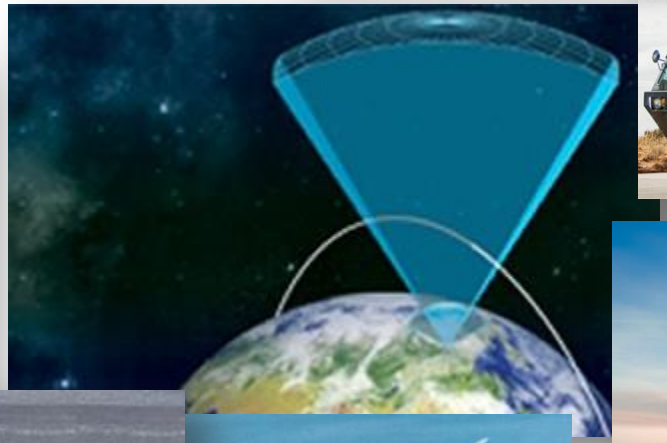
- Different Integration Standards (if any)
- Correlating Information Assets (What goes with What)
- Naming Conventions and Units of Measure
- Accessibility





Raytheon: Who We Are

- Raytheon Business Focus:
  - Defense
  - Aerospace
- More than 72,000 employees worldwide
- More than 40,000 engineers
- 2012 revenues of $26.5B



Raytheon

Getting over the MBE Adoption Mountain

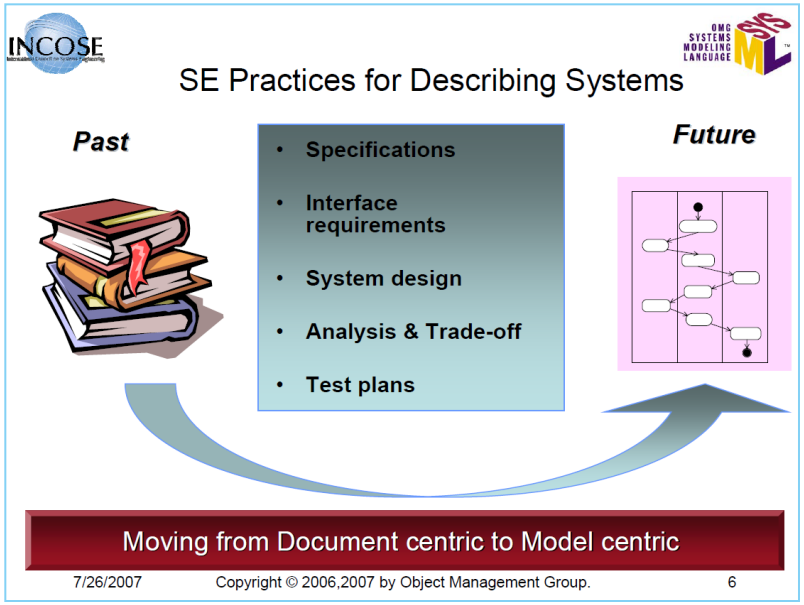
- Address two aspects of this challenge:

1. ***Leveraging automation and data import to accelerate adoption***

2. Leveraging OSLC linking to provide robust multi-disciplinary linking to achieve measurable benefits of MBE quickly

Case Example

- We wanted to apply MBE on a legacy program for which we had over a hundred B5 specifications, each hundreds of pages long, with thousands of interfaces identified
- Migrating all that information from documents to a Rhapsody model by hand would be costly, time consuming, and error prone.
- It would delay “getting over the MBE Adoption mountain”
- Our Goal:
  Extract as much data from these specs as possible and use it to automation generation of a model

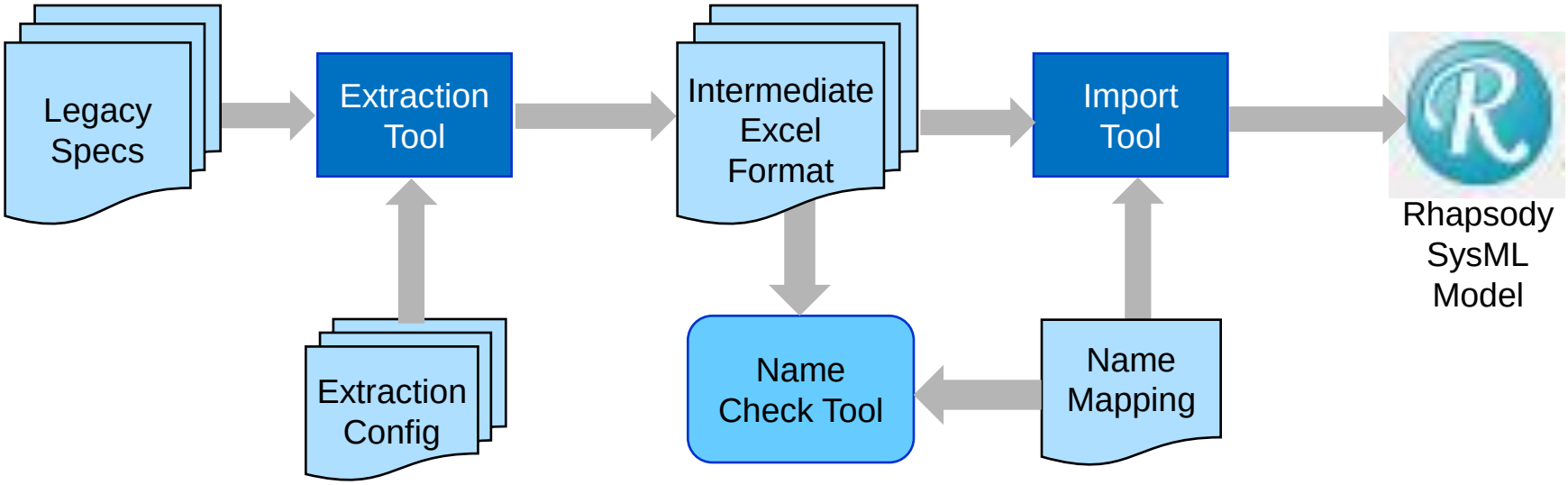


Focus first on the Low-Hanging Fruit

- Our B5 specifications included interface tables, both for internal (component-level) and external (element-level) interfaces
- From one of these simple tables it was possible to extract:
  - Sender and Receiver element and component
  - Message identifiers
  - Message names
  - Message descriptions
  - Several key interface parameters (e.g., rates, data classification)
- We realized that we could also generate:
  - Port names
  - Event reception names
  - Cross references back to the legacy specifications (could be OSLC references to DOORS)
- And that we could later add:
  - Data types (extracted from software artifacts)

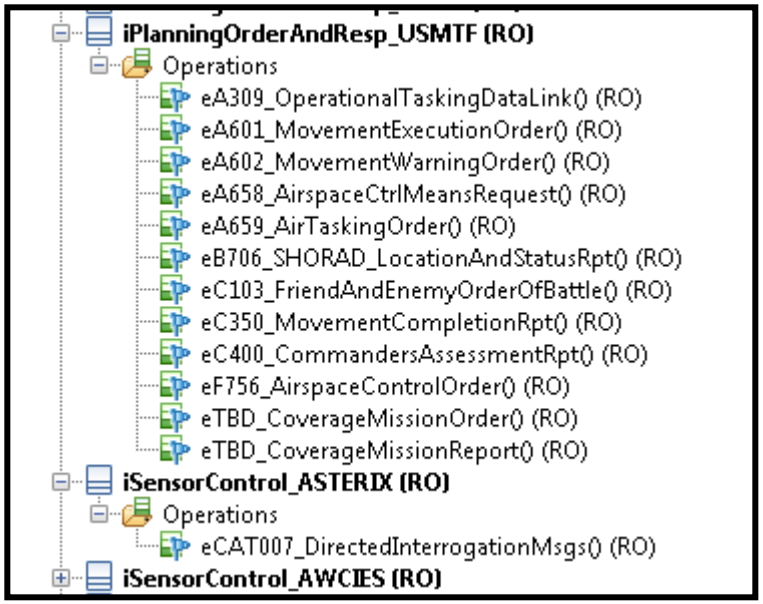
What did we develop?

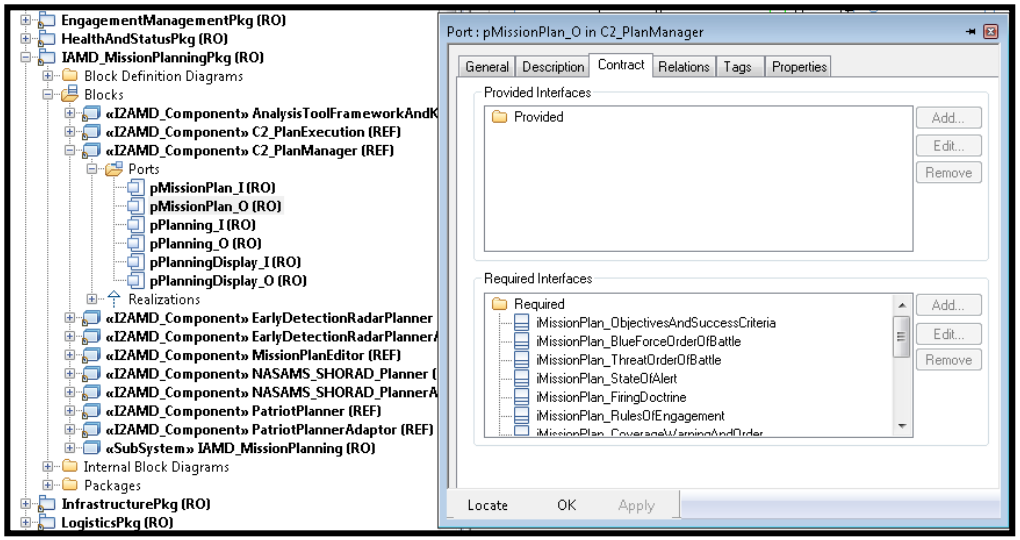
- Process and supporting automation to support transition from legacy Document-Based Systems Engineering to SysML Model-Based Engineering
- Proof of Concept used on a large legacy program:
  - Over 100 legacy B5 Specifications and Interface Description Documents processed to extract interface and structure data
  - Process generated thousands of consistent SysML artifacts in the Rational Rhapsody tool

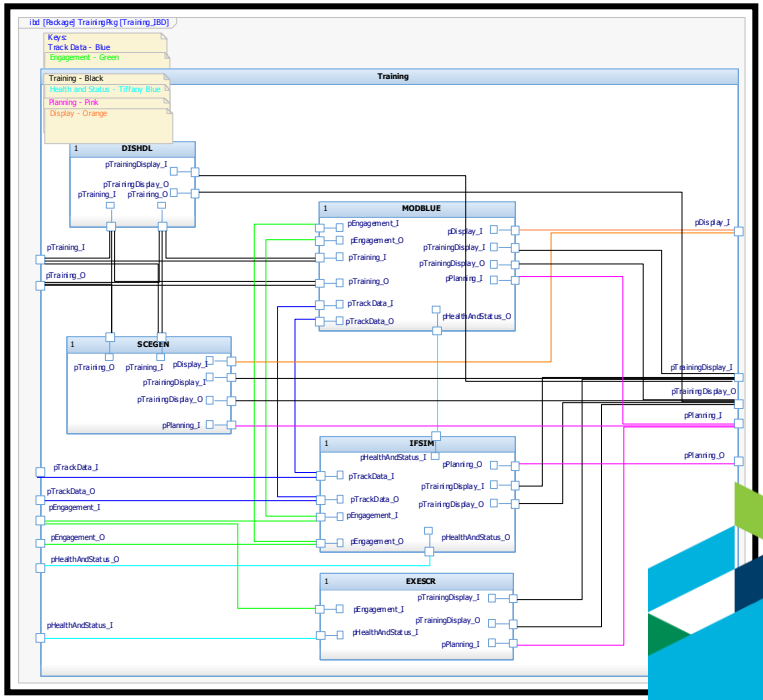


Generated SysML

- Interfaces and Operations on Interfaces
  - Inter-element and component-to-component
- Could have created blocks (but didn't)
- Events associated with interfaces
- Ports on Elements and Components (with Required and Provided Interfaces)
- Receptions on Elements and Components
- Connections between elements and components
- Delegation ports and connections from elements to components
- Relationships among all the above

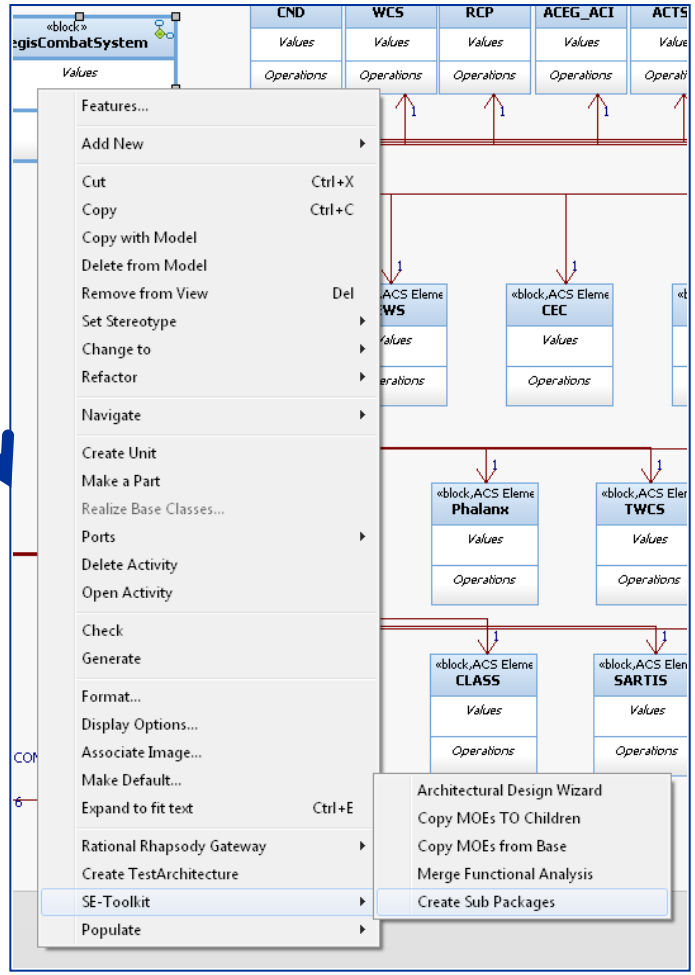






Further Simplify using Rhapsody Plug-ins

- Make the automation ever easier
- Rhapsody Plug-ins are automation accessed through Rhapsody menus
  - Developed using Rhapsody API
  - Rhapsody provided plug-ins (e.g., SE Toolkit)
  - User developed plug-ins
- Creating a product plug-in includes:
  - Writing the Java application
  - Creating a .hep file containing requirements for loading the plug-in
  - Attaching the .hep file to a profile



Getting over the MBE Adoption Mountain

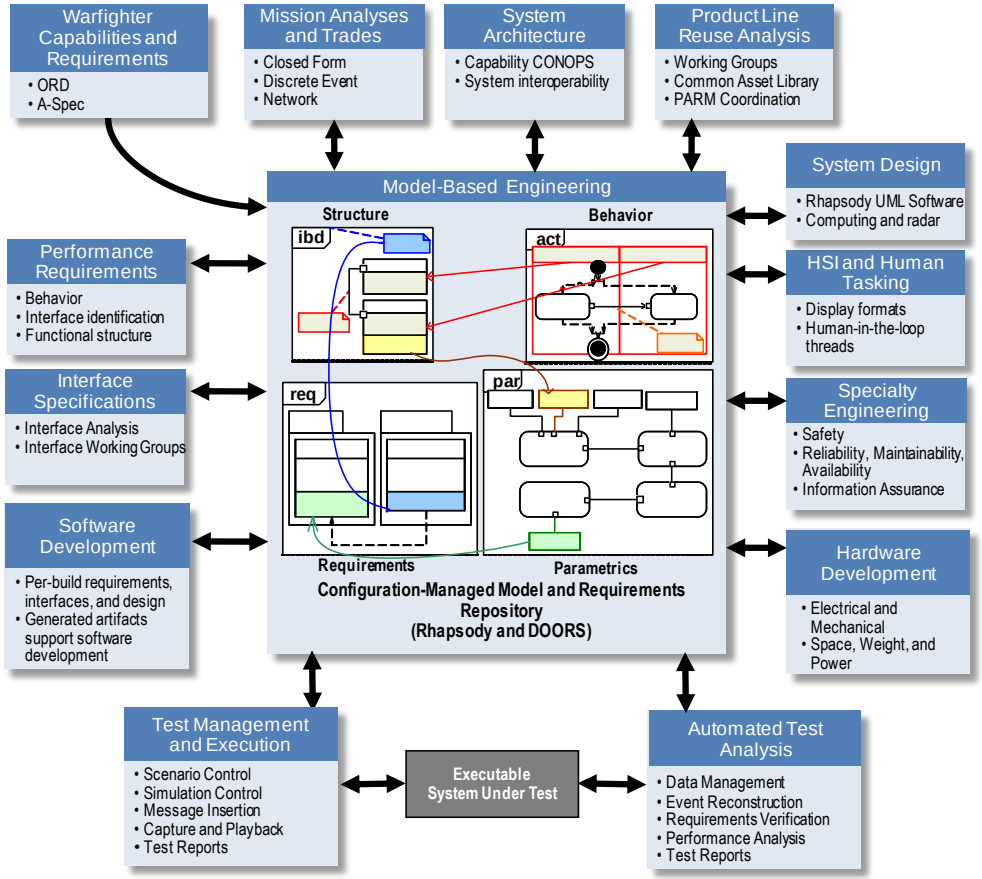
- Address two aspects of this challenge:

1. Leveraging automation and data import to accelerate adoption

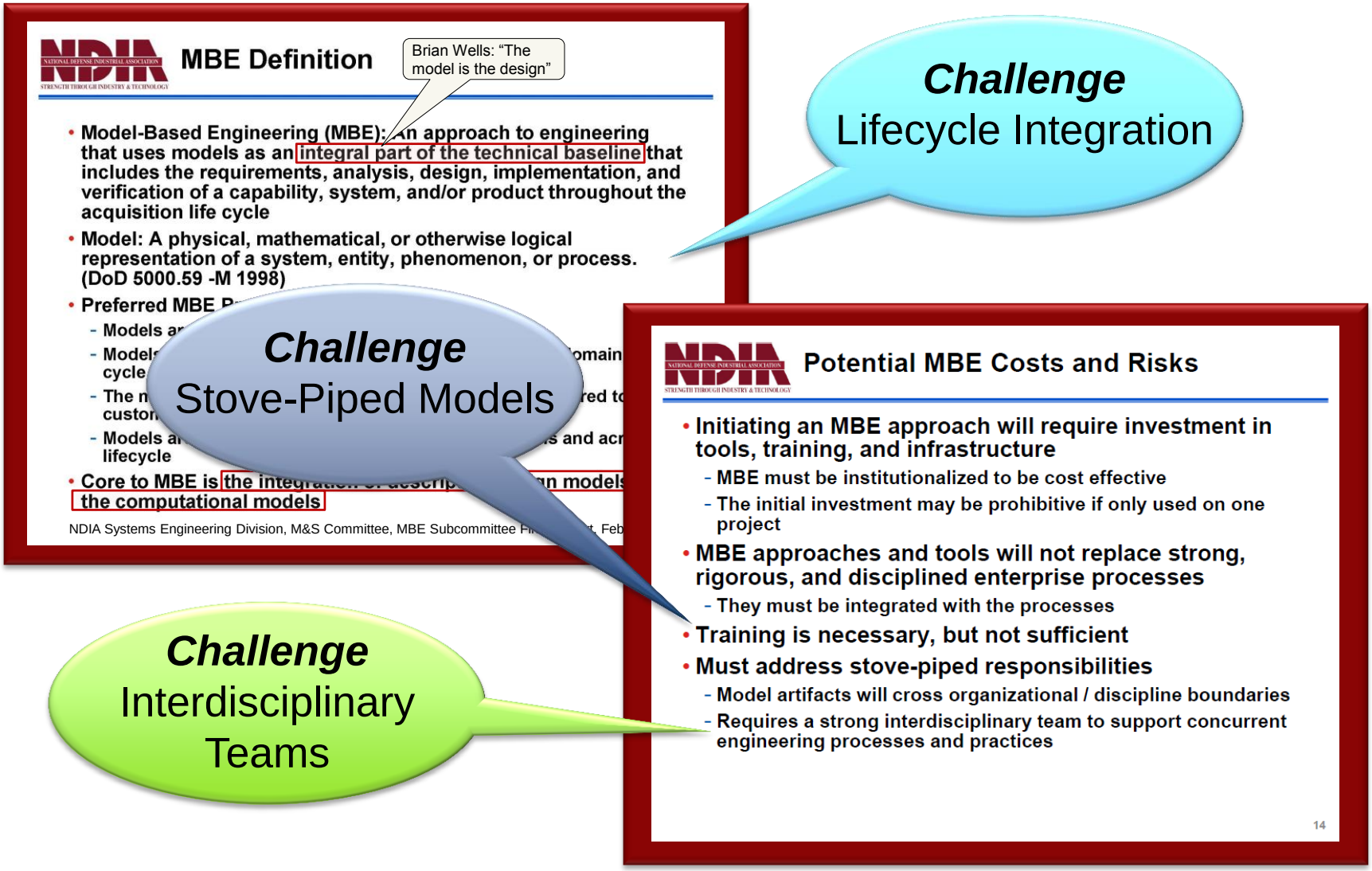
2. ***Leveraging OSLC linking to provide robust multi-disciplinary linking to achieve measurable benefits of MBE quickly***

MBE Demands Multi-Disciplinary Linking

- MBE ~ Model Based Enterprise
- MBE is not one model but rather a collection of multi-disciplinary models
- Many relationships are missing in today's MBE approaches
- Establishing and linking relationships is the key to ensure design consistency
- Changes made from one perspective need to be reflected in others
- Complete / Interconnect Models = Complete / Consistent System Design



Model Based Engineering
Just *Some* of the Challenges

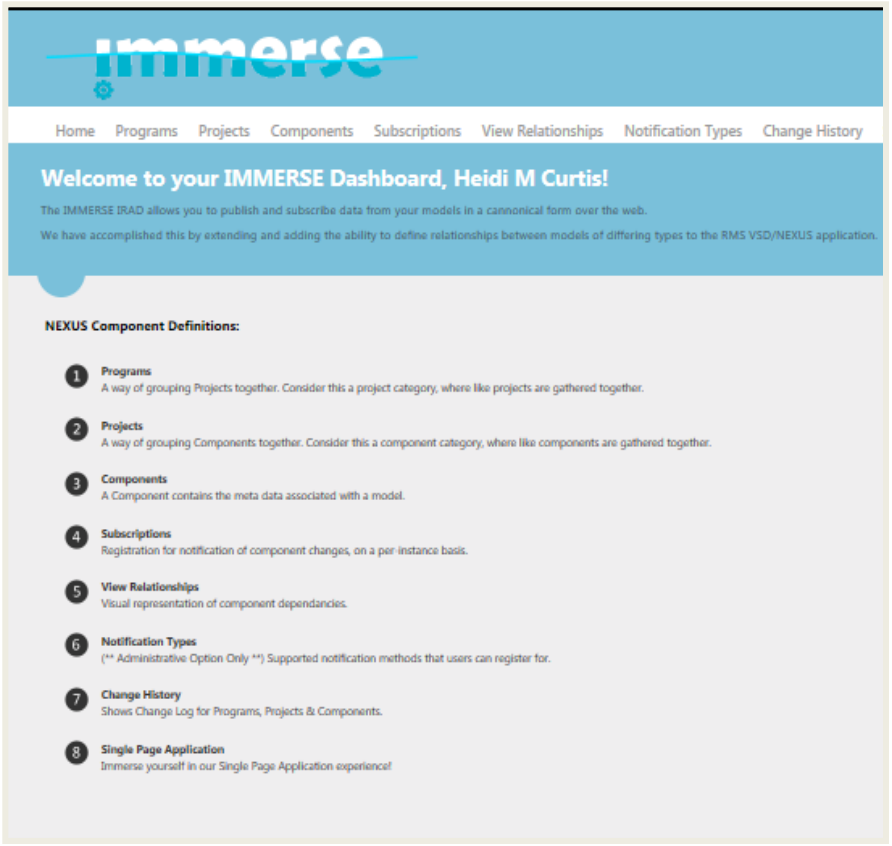


MBE Goals

- **The Models are the Design**
  - From Concept to Mission Scenarios to Architecture to Systems Design to Software/Hardware Design & Development to Verification & Validation and finally to Sustainment
    ….all based on linked Models across engineering disciplines, with end to end semantic consistency
- **Generate Data Once**
  - …and transform and link as needed
- **Compose Future Systems**
  - Model based composition of components, subsystems, assemblies, test artifacts, etc….all linked, query able and managed under configuration control
  - Emphasis on model libraries and patterns for re-use across Product Line Engineering environments

Our Approach - Integrated Models & Methodologies

- Linked Data Architecture
- Tool Agnostic
- Leverage integration standards
  - Open Service Lifecycle Collaboration (OSLC)
  - Resource Descriptor Framework (RDF)
  - Modelica
  - Service Oriented Architecture (SOA)
  - Functional Mockup Interface (FMI)
  - Base Object Model (BOM)
  - Standard for Exchange of Product Data (STEP) / AP233
- MBE Data Model
  - What we want to link
  - Attributes of what we link



Lessons Learned – Model Linking

- Our engineers want to work in their own comfortable modeling environments, so our team needed to provide a (very) user friendly approach to enable linking of cross-domain models
- We could not find a “reusable” data model that would adequately represent our model based enterprise
- Use of RDF as common denominator for model linking
  - Also enables use of many open source sematic tools
- Building plumbing is hard and not very rewarding

Future Directions – Model Linking

- Work with our tool vendors and standards organization to promote multi-discipline model linking
- Continue to experiment through our IMMERSE environment as a means to influence development of commercial solutions
- Working Semantics through technology and process
- Reasoning and Visualization

Summary

- Achieving widespread adoption of MBE in an organization is challenging
  - New processes, tools, and approaches to learn
  - Legacy data to leverage
  - Critical need to show return on investment
- Automation provides a way to help “get over the MBE Adoption Mountain”
  - Get started quickly by building a core set of model artifacts from existing documents
  - Focus on automation to support robust multi-disciplinary data linking to ensure ROI benefits can quickly realized
- Raytheon is focused on improving life cycle productivity through use of MBE methods