Presented to:

National Defense Industrial Association

14th Annual Systems Engineering Conference

U.S. ARMY RDECOM

A Phased Approach to Model Based Capability Development

AMRDEC

Date: October 2011

TECHNOLOGY DRIVEN. WARFIGHTER FOCUSED.

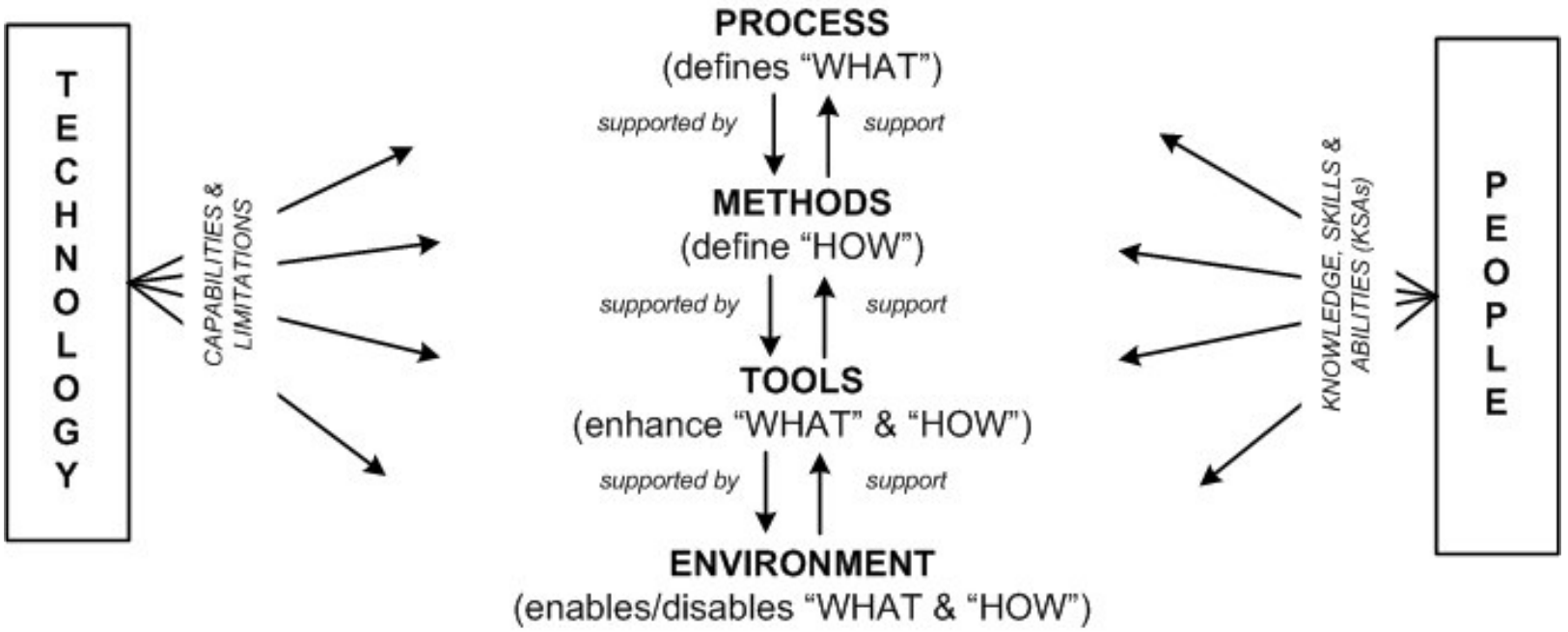
Presented by:

Michael Gully

Systems Engineer

U.S. Army Aviation and Missile Research, Development, and Engineering Center

We've got to find a better, faster, cheaper way to develop systems



The PMTE Elements and Effects of Technology and People.

[Estefan 3]

Better, Faster, Cheaper Systems come from Model Based System Development

- **We're not alone in our beliefs, INCOSE also has a Model Based Initiative**
- **In the INCOSE SE Vision 2020, Model Based System Engineering was 1 out of the 5 main topics.**

Contributors	Inhibitors
• Emergence and maturation of modeling languages and information standards • Continuing evolution of information technology as an enabler of modeling techniques	• Inherent difficulty integrating models across organizational, lifecycle and other boundaries • Limitation of model/data exchange capabilities within modeling tools • Limited MBSE skills

Abstract

The Army Aviation and Missile Research, Development, and Engineering Center (AMRDEC) Software Engineering Directorate (SED) is addressing the Army's need to decrease the time-to-field for new technology and improved systems to meet warfighter needs, while retaining the engineering rigor that is required to meet quality objectives, through the adoption of model-based systems and software development processes, work flows, and tool suites. The gains achievable with the application of model-based methods and tools are well known in the commercial world; however, those environments are typically much less complex, diverse, and constrained than those faced by DoD systems developers. The objectives of the SED model-based initiative are to enhance support to the warfighter by shortening development times, improving system and software reliability, and satisfying increased requirements for system safety (e.g. DO-178), security (e.g. EAL-6), and interoperability certification in the face of shrinking development budgets. This approach has potential for gains in cost, schedule, and developmental efficiencies that could be as high as one or two orders of magnitude over today's increasingly expensive labor-intensive methods.

The SED model-based initiative began with pilot development programs that have successfully applied state-of-the-industry modeling, requirements management, and other design and development methods, tools, and technologies. It is critical to form an effective integration of proper methods, tools, and technologies rather than focus on a single point solution. The SED pilot projects have gathered metrics and documented lessons learned. Systems are now being fielded that have a substantial portion of the software automatically generated from system models at considerable savings. The results of these pilot projects have supported SED's plans for future engineering process improvement, which include plans to institutionalize state-of-the-industry, integrated systems and software engineering processes, work flows, and tool suites to achieve lower system life-cycle costs and shorter time-to-field schedules that cannot be achieved using other formal development/acquisition strategies. These tool suites include not only those for software-intensive systems development in the area of automatically generated software but also those for hardware and complex electronics (programmable firmware).

Institutionalization of these integrated model-based techniques is a daunting task for an organization as diverse as SED just as it is at many other large DoD engineering organizations. This paper discusses our phased approach to achieving a success-oriented, institutionalized, integrated model-based systems and software engineering development culture at SED that informs our already mature processes.

The Model Based Initiative has Clearly Defined Goals

- The SED model-based initiative aims to enhance support to the warfighter in the face of shrinking development budgets.
  - Shortening development times
  - Improving system and software reliability
  - Satisfying increased requirements for safety (e.g. DO-178),
  - Satisfying increased requirements for security (e.g. EAL-6)
  - Satisfying increased requirements for interoperability certification

- The SED model-based initiative began with pilot development programs that have successfully applied state-of-the-industry modeling, requirements management, and other design and development methods, tools, and technologies.

- This paper discusses our phased approach to achieving a success-oriented, institutionalized, integrated model-based systems and software engineering development culture at SED that informs our already mature processes.

The Model Based Initiative Has Been Defined and Bounded

Need:

Better, Faster, Cheaper Systems Development

Objective:

Change the way DoD systems are developed from a document centric paradigm to a model centric paradigm.

Operational Concept:

Phase I: Stovepipe Technology Refinement. **Phase II**: Integrate Stovepipe Technologies. **Phase III**: Proof of Concept Pilot Projects and Measure Results. **Phase IV**: Compare/ Analyze Pilot Projects Against Traditional Methods. **Phase V**: Improve Model Based Methods, Document Processes, Train People.

Assumptions:

Integrated tools provide users the ability to tailor their selections to those most applicable to the problem they are trying to solve. True Institutionalization is only achieved through esprit decorps.

Constraints:

Resources for the initiative are extremely limited. Progress and contributions to the initiative are made in addition to normal duties. Personnel working on the initiative are project funded. Work or integration of the Model Based Initiative must not unduly add risk to projects in a way which might inhibit their success.

Authority and Responsibility:

The contributors have been authorized to pursue the Model Based Initiative by AMRDEC SED Management provided they remain compliant with the assumptions and constraints above.

Kurt Lewin's Change Management Model

- **Kurt Lewin is recognized as the "founder of social psychology".**
- **Interest in groups led to research focusing on factors that influence people to change, and three stages needed to make change successful.**

- **Unfreeze -** This first stage is about preparing ourselves, or others, before the change (and ideally creating a situation in which we want the change). It involves getting to a point of understanding that change is necessary, and getting ready to move away from our current comfort zone.
- **Transition (Change) –** This second stage occurs as we make the changes that are needed. Using role models and allowing people to develop their own solutions can also help to make the changes. It's also really useful to keep communicating a clear picture of the desired change and the benefits to people so they don't lose sight of where they are heading.
- **Freeze (Refreeze) -** This stage is about establishing stability once the changes have been made. The changes are accepted and become the new norm. People form new relationships and become comfortable with their routines. This can take time.

Private Industry is a Different Environment than DoD

- Private industry widely uses model-based development tools such as IBM's Rhapsody
  - Examples: automotive, personal electronics, and telecommunications industry
  - They are successful using the tool for their product development
- The use of these tools in those environments rarely involves the DoD need to:
  - Support extended "use and deployment" life-cycle phases of decades
  - Require formal documentation
  - Require formal developmental, verification, qualification testing
  - Generate extensive artifacts to validate for safety- and IA/security-critical applications required to meet DoD standards (DO-178, DO-254, EAL-6, etc)
  - Require integration of sometimes conflicting domain areas
- Thus Model Based System Development (MBSD) technologies are not mature in the DoD environment even though the are widely used successfully in private industry

Model Based Tools are at TRL 4 (roughly)

- **We estimate the current set of Model Based System Development tools (as a "set of capabilities") is at TRL 4 based on informal internal assessments of their application to the DoD environment. [Special Report]**
  - **Its not just using tools, its using integrated tools.**
- **The Model-Based Development for Defense systems is at TRL 3 or 4**
  - **TRL 3 Analytical and experimental critical function and/or characteristic proof-of-concept:** Proof of concept validation. Active R&D is initiated with analytical and laboratory studies. Demonstration of technical feasibility.
  - **TRL 4 Component/subsystem validation in laboratory environment:** Standalone prototyping implementation and test. Integration of technology elements. Experiments with full-scale problems or data sets.
- **The goal of the SED Model Based Initiative is to elevate to TRL 9**
  - **TRL 9 Actual system "mission proven" through successful mission operations (ground or space):** Fully integrated with operational Hardware/software systems. Actual system has been thoroughly demonstrated and tested in its operational environment. All Documentation completed. Successful Operational experience. Sustaining engineering support in place.

[NASA TRL Definitions]

DoD Acquisition is Evolving

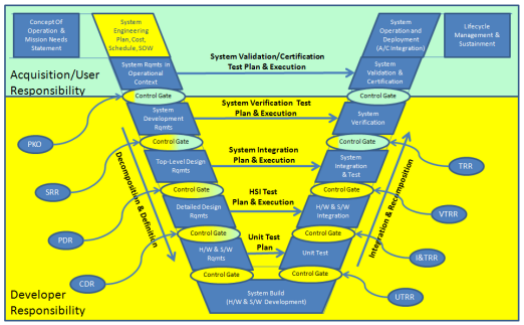
- DoD Acquisition Program (and Project) reviews have evolved over time to reflect what is state-of-the-art.
  - Paper Documents and Engineering Notebooks reflected what systems were
  - Advances in IT enabled the creation of "electronic" documents. Often, the only official copy of these documents is a signed "hardcopy" for decision milestones and gate reviews.
  - Technology has advanced to a point where the technology can easily support fully electronic or model-driven programmatic reviews. [Estefan 6]

- Department Of Defense Architecture Framework (DoDAF) is an attempt to enable these model based programmatic reviews. DoDAF isn't intended to provide the modeling needed to develop systems. It's intended to enable system developers to use modeling and roll the data in those models up for programmatic review.
  - "Fit-for-Purpose Views' are user-defined views of a subset of architectural data created for some specific purpose…This enables organizations to use their own established presentation preferences in their deliberations."

Model Based Technology Applies to the Whole System Lifecycle

What Areas of the System Lifecycle Does it Address? All of them.

- Model Based Development and Engineering does not replace an organizations need for process nor the use of lifecycle models.
- Attempting to replace an organization's processes with a model based methodology will have negative results.
- Instead, it is suggested that the Model Based Approach be adopted and tailored to fit an organization's existing processes.
- Looking at the diagrams of the some Traditional Life Cycle Models and the Model Based Development Life Cycle Models their similarities (and differences) are immediately apparent.
- Example: compare a standard "V" with the corresponding Rational Harmony SE © "V"



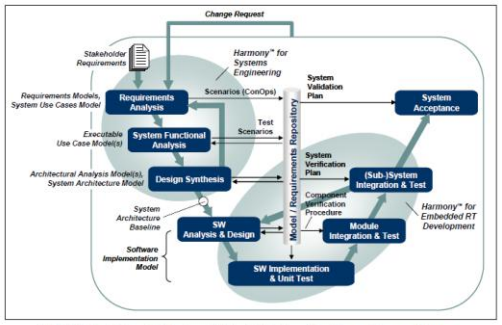
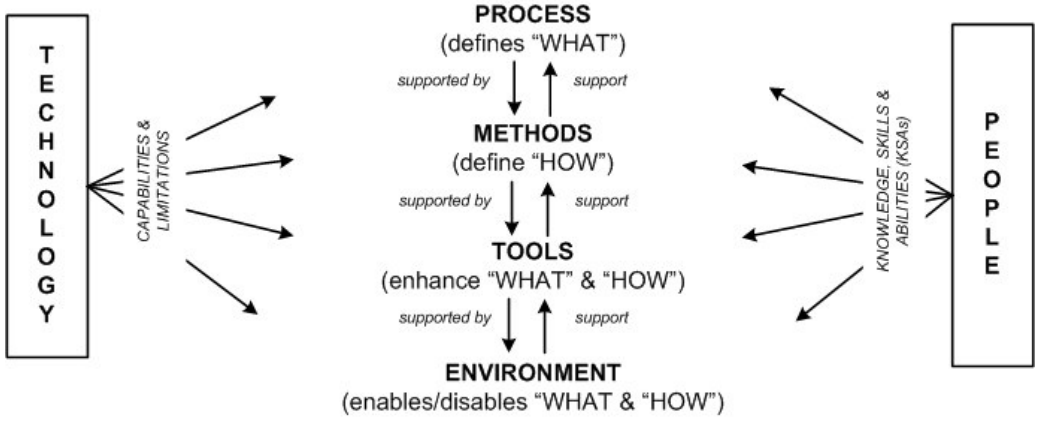


Fig.2-1 Rational Integrated Systems / Embedded Software Development Process Harmony

Our End State is Clearly Defined

- **Our end state is a well-defined, seamlessly integrated enterprise approach to use for the development and maintenance of systems.**



- **Our phased approach includes:**
  - **The identification of the requisite tools and associated methods and capabilities must be integrated into our end state**
  - **The maturation of each of those tools/methods/capabilities and integration of them into the others at the correct place**
  - **The iterative use and refinement of the integrated whole until it meets the MOEs/MOPs required for our system development infrastructure**

There are Defined Phases to Achieving Our Desired End State

- **Phased Approach:**
  - **Phase I: Identify Tools, Processes, Methods, Environment**
  - **Phase II: Stovepipe Technology Development**
  - **Phase III: Integrate Stovepipe Technologies**
  - **Phase IV: Proof of Concept Pilot Projects and Measure Results**
  - **Phase V: Compare/ Analyze Pilot Projects Against Traditional Methods**
  - **Phase VI: Improve Model Based Methods, Document Processes, Train People**

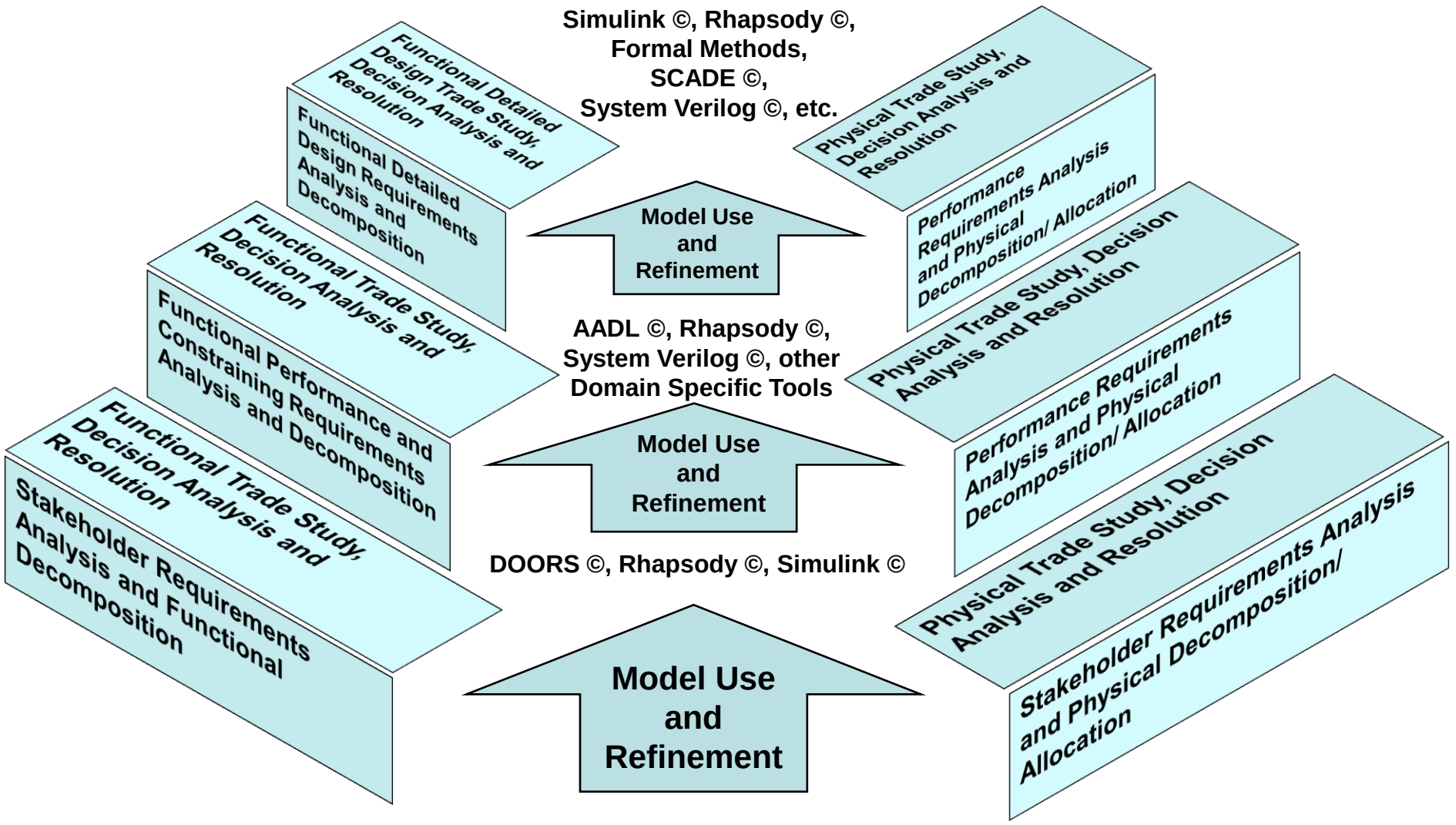
- **We're only starting Phase III now but by refining the model based technologies in stovepipes (Phase I-II) we have reduced the time identifying candidate tools for the various functions.**

Phase I: Identify Tools, Processes, Methods, Environment

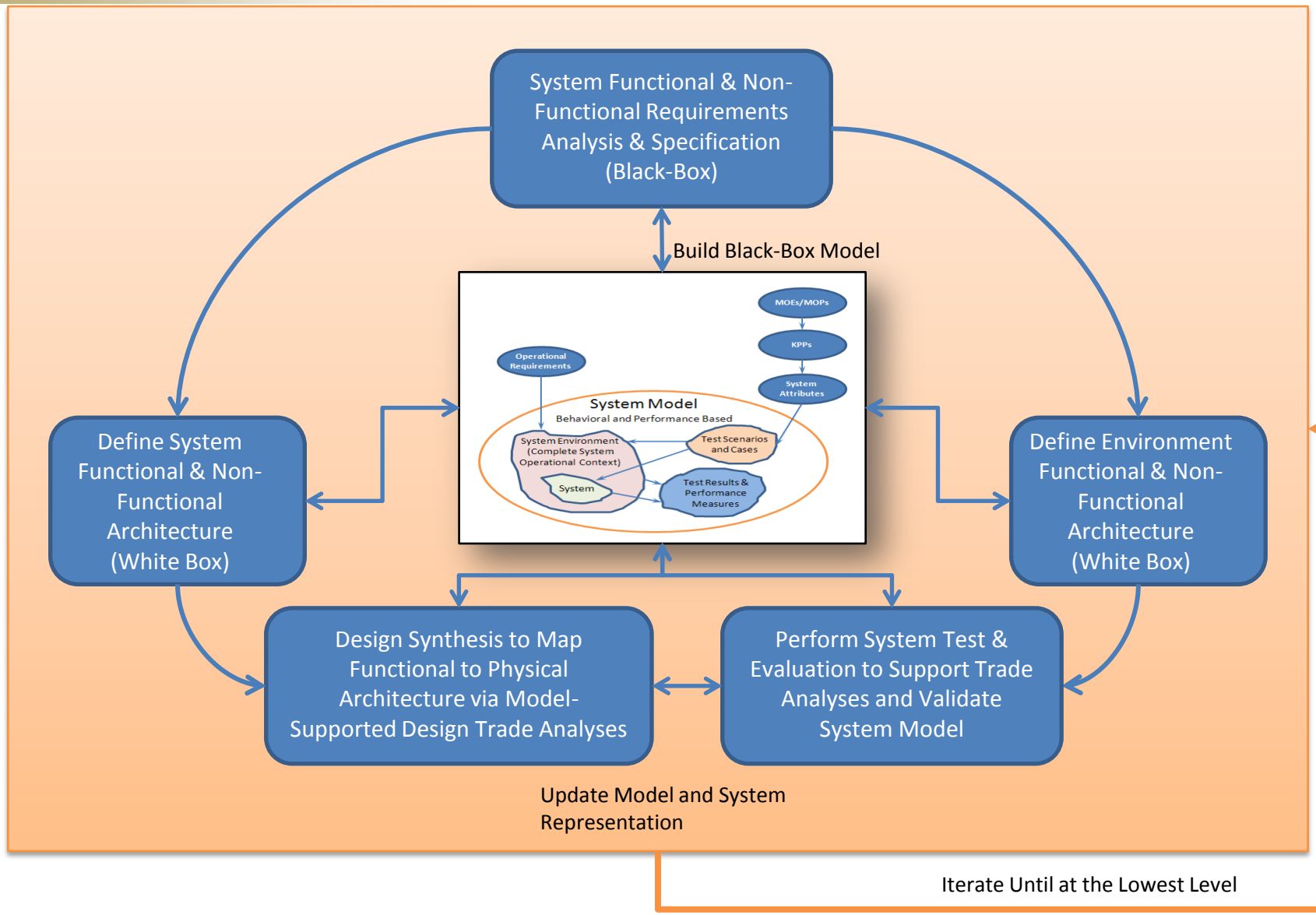
- **Performance (mainstream of system development and what is normally considered in "Model-Based System Development")**
  - **Requirements Analysis & Specification**
    - DOORS ©
  - **Functional Analysis & Design Synthesis**
    - Rhapsody ©, Enterprise Architect ©, AADL ©, Simulink ©, others
  - **Performance Modeling**
    - Simulink ©, many other domain-specific
  - **Physical Architecture Analysis & Design Synthesis**
    - AADL ©, Simulink ©
  - **Software Development**
    - Rhapsody ©, Simulink ©, others
  - **Formal Language Modeling for Safety Critical and Security Critical**
    - SCADE ©, other domain specific – e.g. those for Complex Electronics
  - **Hardware/Complex Electronics Modeling**
    - Xilinx ©, Altair ©, Simulink ©, many others

- **Other Specialized Domain Specific Areas and Tool Functions**
  - **Cost Modeling for Production to support AUPC, etc**
  - **Production Modeling**
  - **Reliability Modeling**
  - **Network Modeling**
  - **Security/IA Compliance Verification**

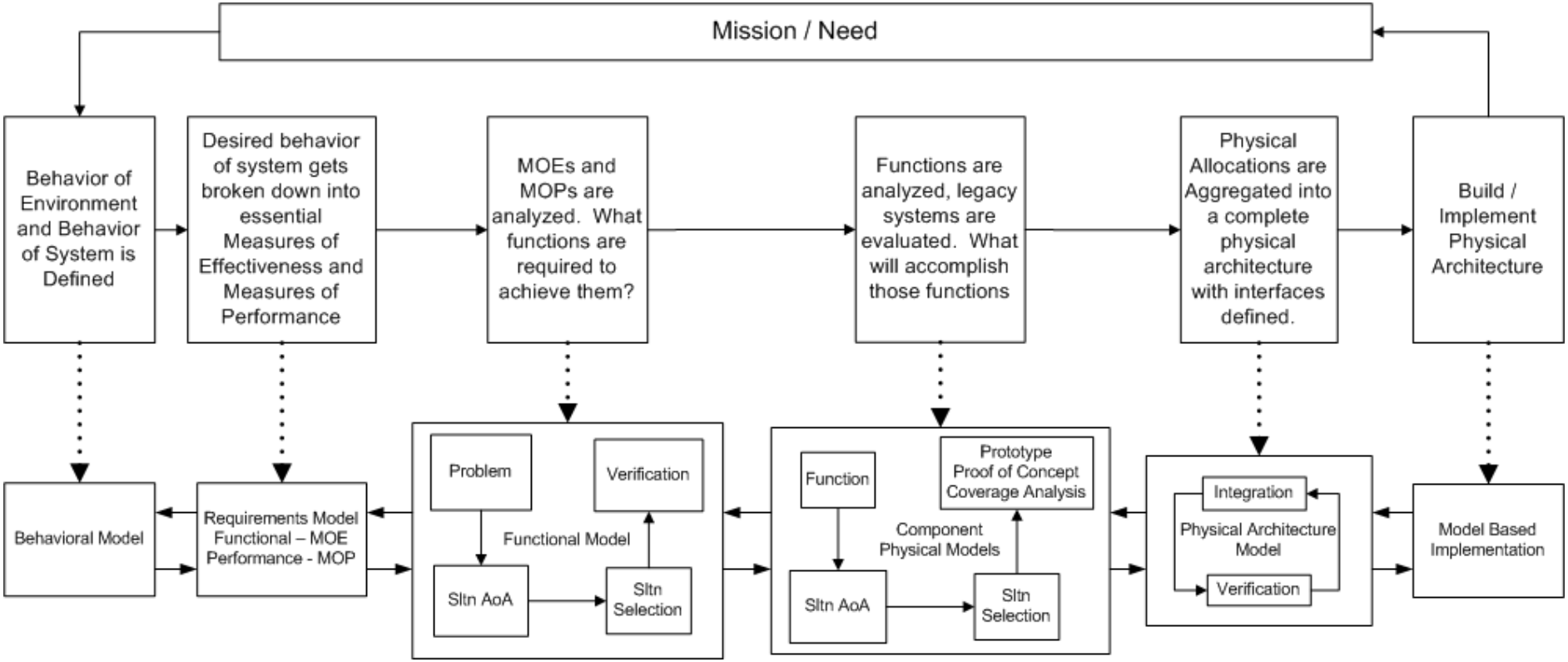
Different Tools Are Available as the System Model Gets Refined



The System Model Connects with Every Aspect of Development



The "System Model" isn't One Model



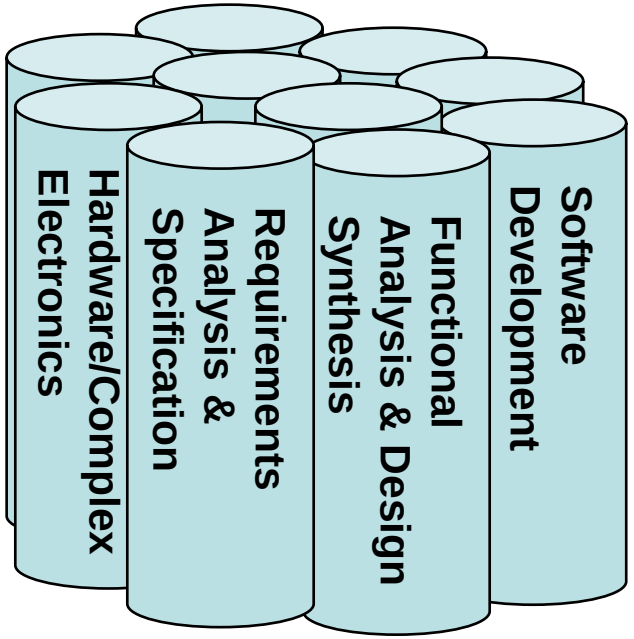
Phase II: Stovepipe Technology Development

- Phase II of the SED Model Based Initiative required buy-in from customers.
- They enabled us to “unfreeze” our traditional development approach and pursue a Model Based approach.
- Required the customer to learn in order to become familiar with modeling languages
  - Project tasks on the schedule look different than normal
  - Code comes at the very end of the project
  - Pictures have real meaning
  - Models take the place of documents for design

- We encouraged customers to only adopt the model-based paradigm for specific functions of their projects in order to drive down cost and mitigate risks associated with technology insertion.

Phase II: Stovepipe Technology Development

- We Use Models to:
  - Communicate Behavior
  - Make/ Verify Design Decisions
  - Support Functional to Physical Requirement Allocation
  - Define and Impose Constraints and Performance Requirements
  - Make Decisions between S/W, H/W, Complex Electronics



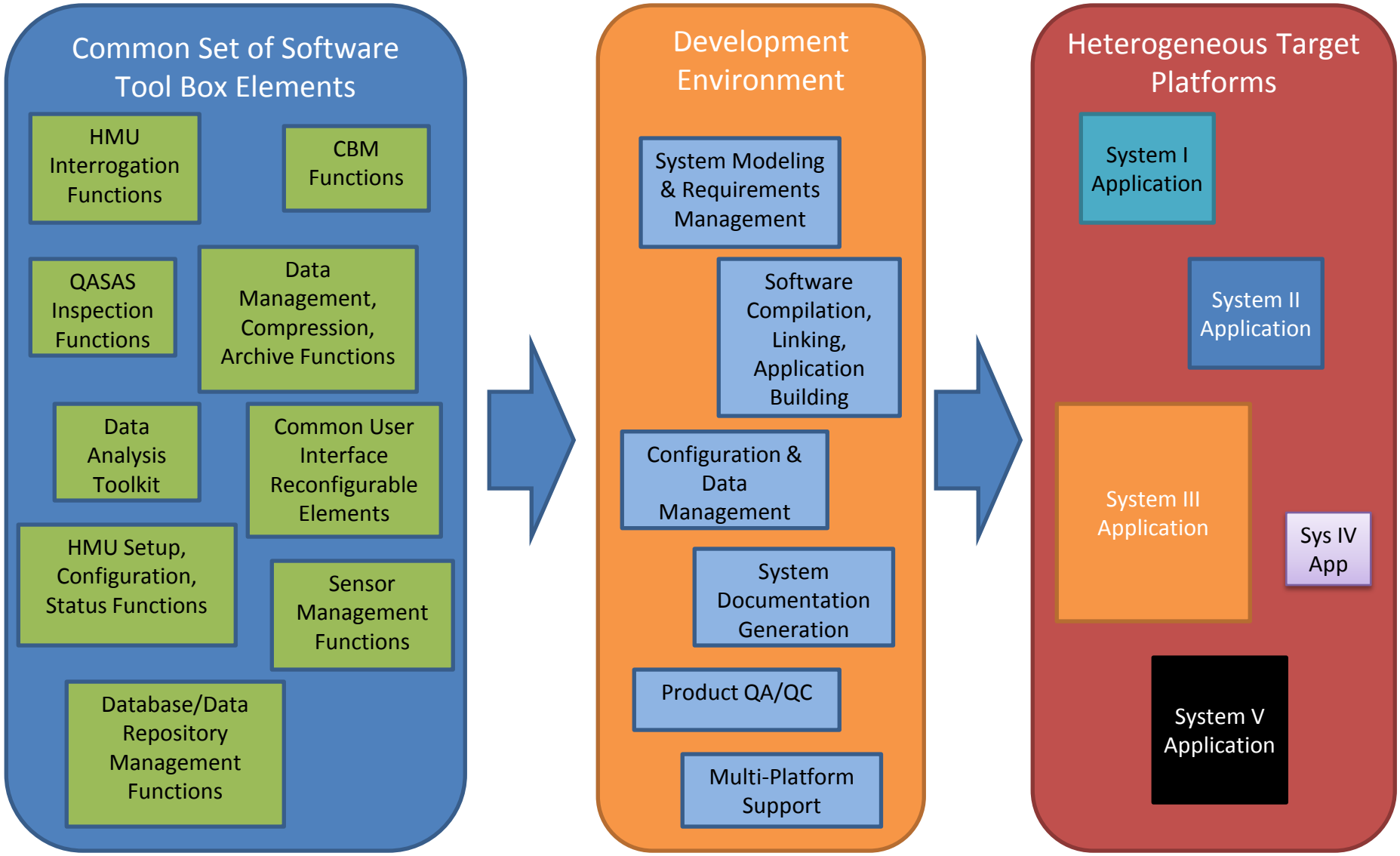
However:

- Engineering disciplines use different tools
- No tool is comprehensive (nor should be)
- Tools don't integrate

Pilot Project A: Lessons Learned

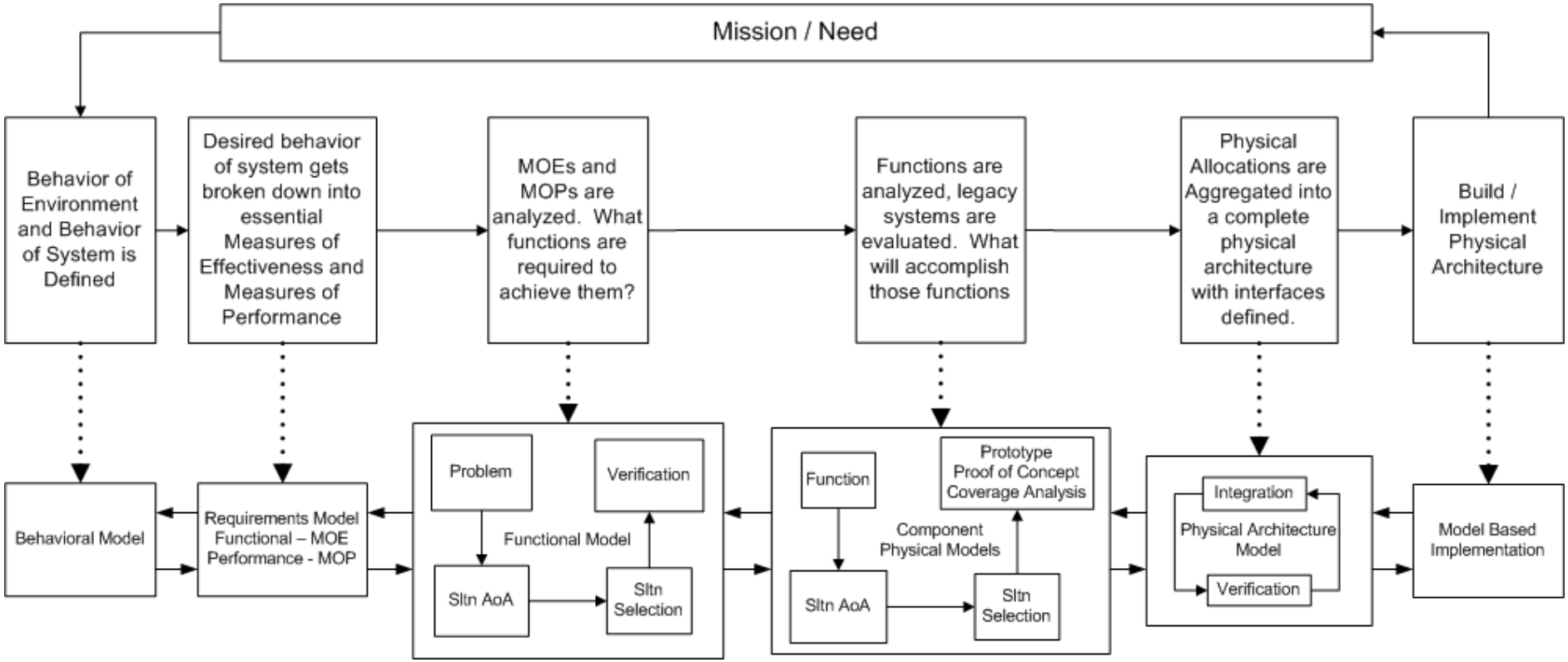
- Concurrent development and implementation of the Test Environment model saves time by identifying errors before they can be propagated.
- Error propagation is mitigated early, (even during requirements analysis) using concurrent, Model Based Testing to drive SUD model
- Work flows provide the capability to standardize work products
- Don't attempt this without training
  - Even with training, continued mentoring is vital
  - Training is necessary but not sufficient
  - This approach may not be cost effective if it is not institutionalized (cost may be prohibitive if only used on one project)
- Must integrate model-based development activities into standard enterprise system engineering – Tools and Workflows don't replace system engineering processes.
- Independent SUD functional model development per use case followed by integration of models is labor intensive
- Time is saved when transitioning from Systems Engineering to SW Engineering by using a common modeling tool suite and language (SysML & UML)

Pilot Project B Highlights a Model Based Approach to Reuse



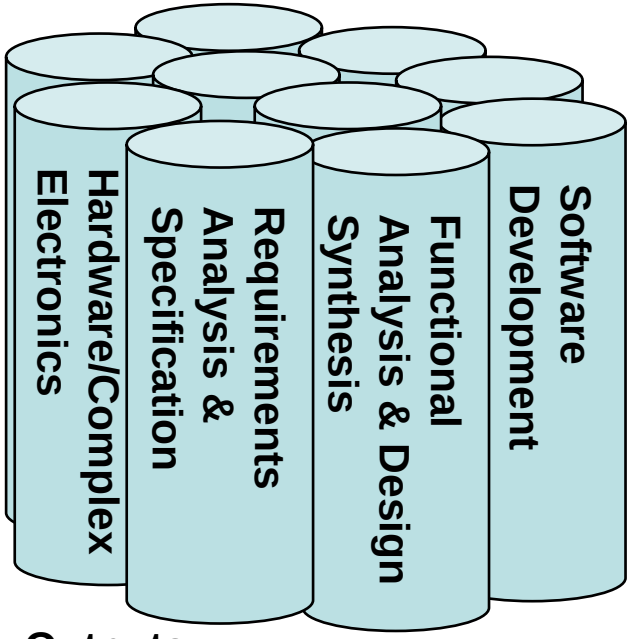
Phase III: Integrate Stovepipe Technologies

- **Our Initiative to Integrate the Technologies is being approached no different than any other development effort**



Phase III: Integrate Stovepipe Technologies

Inputs:
Process, Methods, Tools, Environment, People



Outputs:
Varied Quality
Repeatable (with same team)
Predictable (with same team)
Often Reduced Time to Field
Partial Reusability

SED Model Based Initiative

Inputs:
Process, Methods, Tools, Environment, People

Integrated Model Based Engineering Solutions for DoD System Lifecycle Support

Outputs:
High Quality
Repeatable (Organizationally)
Predictable (Organizationally)
Drastically Reduced Time to Field
Comprehensive Reusability

The Initiative is Success Oriented

- Being “Success Oriented” means using the technology to solve the problems we face today.
- The needs of the warfighter cannot wait for a pure R&D effort
  - Technology insertion adds risk to projects and the quality of products that support the warfighter cannot be compromised.
- The answer is to piggy-back the technology insertion on pilot projects which still utilize traditional development methods.
- Parallelization of traditional and model-based methods has benefits:
  - Comparison (verification) of work-products
  - Metrics to analyze differences
  - Mitigates Risk of technology insertion
- Using Pilot projects means that over-time the set of capabilities developed will be integrated, comprehensive, and proven to adequately support our industry needs.

In Conclusion, We're in a State of Change

- DoD needs the Model-Based Initiative
  - Better systems are configurable, expandable, and maintainable
  - Faster transition from technology to systems in the field
  - Cheaper cost through every stage of the product life cycle
- Buy-in from customers enabled the Model-Based Initiative
- Our stove-piped model based technology refinement has yielded success and lessons learned.
- Our successes have increased awareness (and interest) from our customers, they request that we use this technology for their development programs.
- We're approaching the next phase as a formal system development
  - Characterizing the required behavior
  - Dividing behavior into functions
  - Identifying component tools to fulfill those functions
  - Integrating those tools into a consistent whole
  - Use the integrated set, institutionalize though esprit decorps

References

- DoDAF Architecture Framework Version 2.02. DoD Deputy Chief Information Officer. August 2010.
- Estefan, Jeff A. “Survey of Model-Based Systems Engineering (MBSE) Methodologies”. Rev B. June 2008. INCOSE MBSE Initiative.
- http://www.change-management-coach.com/kurt_lewin.html
- NASA Definition Of Technology Readiness Levels (used by various DoD elements such as DHS)
- Osborne, Leon. *Clarus Concept of Operations*. Publication No. FHWA-JPO-05-072, Federal Highway Administration (FHWA), 2005.
- SPECIAL REPORT CMU/SEI-2002-SR-027: Using the Technology Readiness Levels Scale to Support Technology Management in the DoD’s ATD/STO Environments
- Systems Engineering Best Practices with the Rational Workbench for Systems and Software Engineering. Deskbook. Release 3.1.1. Model-Based Systems Engineering with *Rational Rhapsody* and *Rational Harmony for Systems Engineering.* July 2010.

Backup Slides

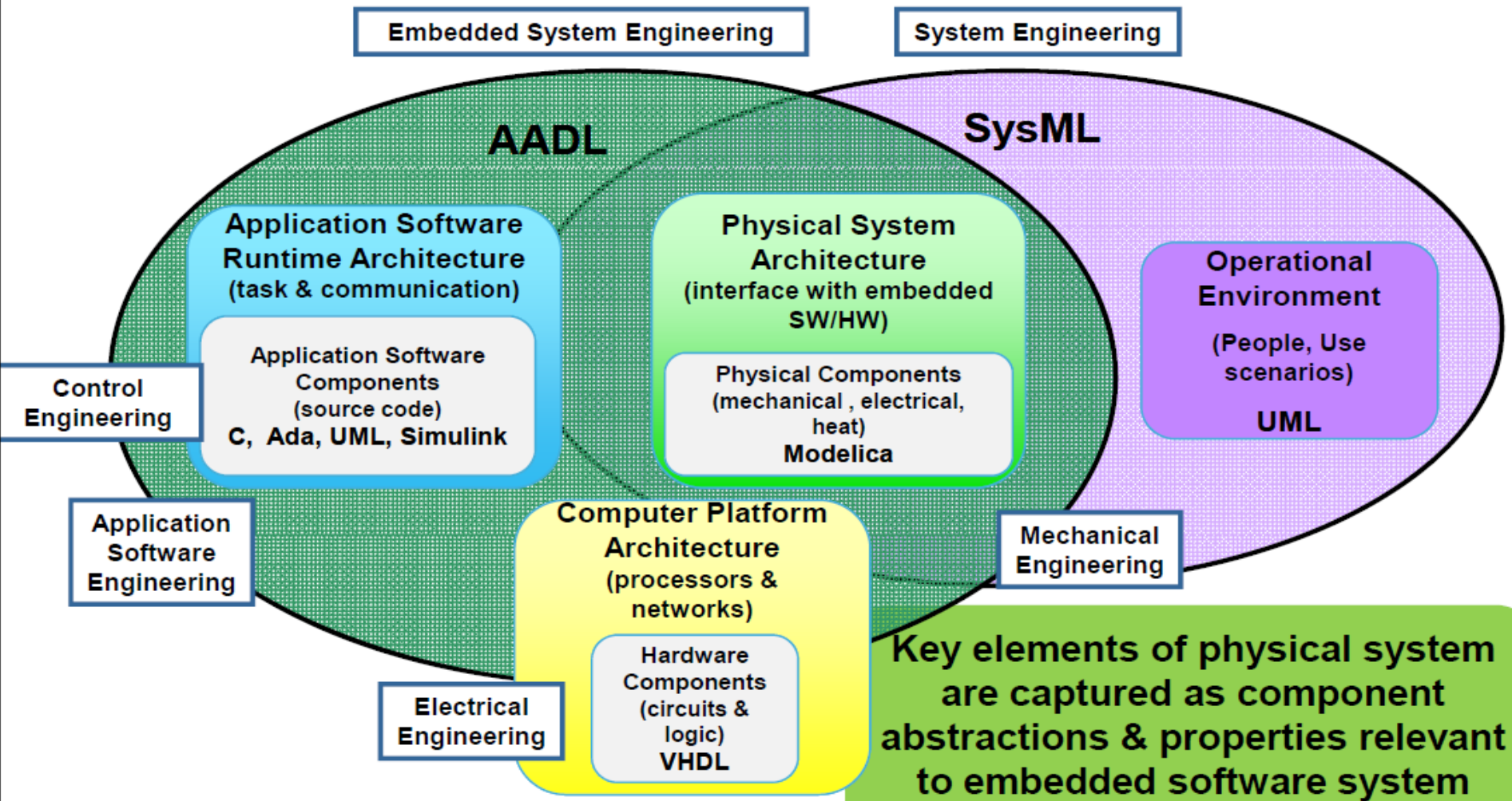
Why the AMRDEC SED?

- The AMRDEC SED is an ideal location for Model Based maturation
  - SED has projects that span the full spectrum of Army Strategic and Tactical capabilities
    - Such a wide variety of Strategic and Tactical capabilities requires a very diverse set of knowledge.
    - Not only are their Hardware, Software, Complex Electronics but within each of those functional domains are Aviation, Communications, Missiles, Rocket specialties.
  - As a Life-Cycle support center, SED is familiar with product characteristics that lead to maintainable systems
    - The systems developed at SED often stretch from Pre-Milestone A all the way through Sustainment.
    - Proving the Model Based design/development paradigm at such a location provides assurance that it will work throughout the Army.
  - SED is capable of bridging the gap from Model Based Systems to Model Based Software Engineering

Architecture Analysis and Design Language (AADL)

- *Architecture modeling language*
  - D*eveloped by and for* avionics, aerospace, automotive, and robotics communities.
  - Conforms to *SAE International standard document AS 5506A*
  - Supports *model-based engineering concepts for embedded real-time systems*
  - *Contains* language and tools to
    - model
    - analyze
    - generate
- Uses component-based notation for the *specification of task and* communication *architectures*
  - *real-time*
  - *embedded*
  - *fault-tolerant*
  - *secure*
  - *safety-critical*
  - *software-intensive systems*

Architecture Analysis and Design Language (AADL)



Key elements of physical system are captured as component abstractions & properties relevant to embedded software system analysis

Simulink for Model Based Design

- Very comfortable with the maturity of the environment
- Used for system modeling and simulation
  - Complex hydraulic control system and mechanical positioner modeled and simulated
    - With some effort, integrated a high-fidelity dynamics model of the mechanical system from a third-party, Motion Genesis, was integrated with Simulink
    - With a single button push, we are able to produce a C/C++ model of the system that can be used for software development
  - Models developed used as the basis for re-implementation of the control law to improve system responsiveness and to target a new hardware platform
    - Moving from a processor-based control law to an FPGA-based, processor-free control law implementation
- Verification of the models is a significant undertaking
  - Models have been partially verified, but some second order effects are not tracking well. Additional work is being performed to identify and correct discrepancies in the model and the actual system response
  - The C models generated for the control law are sufficient to use in an tactical environment but ready for human-driven life-cycle support
  - FPGA RTL is not readily mapped to the desired architecture, so the models are used for verification of the human-developed RTL code

FPGAs for Obsolescence Mitigation

- FPGAs provide a way to ensure program-specific Intellectual Property (IP) from technological obsolescence
  - All program-specific IP is being developed by SED engineers to provide designs for SED client agencies with no IP licensing or proprietary data rights restrictions
  - All IP is portable between FPGA vendors with minor exceptions such as on-chip Ethernet MAC and similar high speed interfaces
- FPGAs have replaced all other components as the IC process leader
  - SDRAMs once held this position, but has been supplanted by FPGAs (Xilinx/Altera) as the preferred technology driver for IC process development
  - This development positions fabless vendors (Xilinx/Altera) as technology drivers and ensures their access to fabs

Institutionalization

- Get Buy-In from Stakeholders (Users, Customers, Management)
- Institutionalization must come from
  - The Top Down (Customers and Management)
  - The Bottom Up (Users)
  - The Left Side (Functional Leads)
  - The Right Side (Project Leads)
- The bottom up is achieved by:
  - Successful projects
  - Development of and maintenance of a cross-functional Team
  - Adding new people to the Team and releasing others to evangelize
- The top down is achieved by:
  - Encouragement of learning new techniques
  - Providing Flexibility for deliverables
- The left side is achieved by:
  - Infrastructure support
- The right side is achieved by:
  - "Buying-in" and using it