Introduction to the Architecture Centric Virtual Integration Process (ACVIP)

Rand Whillock
Adventium Labs
Rand.Whillock@adventiumlabs.com

This material is based upon work supported by the U.S. Army Development Command (DevCOM), Aviation & Missile Center (AvMC), Technology Development Directorate for Aviation (TDD-A), ATTN: Security Office, Bldg 401, Lee Blvd, Fort Eustis, VA 23604-5577 under contract no. W911W6-17-D-0003. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the U.S. Army DevCOM.

Adventium LABS

ACVIP Modeling and Analysis Training Overview

- Today's meeting will be an overview of the first module of the ACVIP Modeling and Analysis training being developed by Adventium Labs.
- This will be an interactive session.
- Some of the discussions may need to be moved offline so we can keep to a schedule and show you the full module.
- We encourage participants to turn on their cameras and participate in the exercises.

ACVIP Modeling & Analysis Training

An interactive hands-on training course to give participants an understanding of the **Architecture Centric Virtual Integration Process (ACVIP)**, and to provide hands-on experience with key modeling techniques and analysis tools.

ACVIP includes modeling methods, processes, and analysis tools that can be applied to embedded computing system design projects to:

- Reduce technical risk
- Reduce cost and schedule overruns
- Support conformance to industry standards & required certifications

Course Details

- Interactive, hands-on Open Source AADL Tool Environment (OSATE) and Architecture Analysis & Design Language (AADL) modeling and analysis
- Live, virtual class via Teams
- Based on the ACVIP Modeling and Analysis Handbook
- Target audience: System designers and developers
- 1st session (overview) will be useful for project managers
- Experience with AADL useful but not required
- 8 sessions lasting about 60 - 90 min each
- Classes begin March 2021

Course Outline

1. Benefits of ACVIP
2. Developing an ACVIP Management Plan
3. Structuring Models for Delivery and Virtual Integration
4. Defining the Model Content Required for Analysis
5. Incremental Design, Analysis, and Resolution
6. Model Based Development
7. Model Based Integration
8. Assuring Model Conformance & Certification

Agenda

- Module Objectives
- What is ACVIP?
- Project Management Issues
  - Example Story
- Project Management Goals
- ACVIP Solutions
- Exercise: Match issues with ACVIP solutions
- Example ACVIP Work Flow Walkthrough

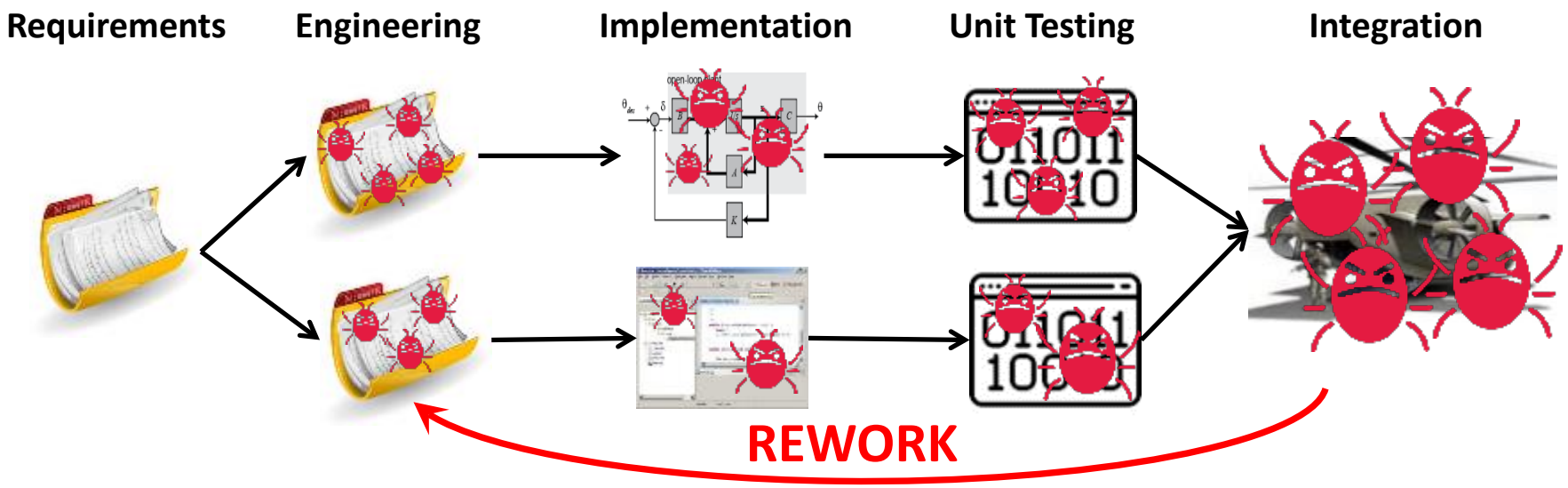
Learning Objectives

- Understand how ACVIP can address many of the issues found during complex embedded computing system development programs.

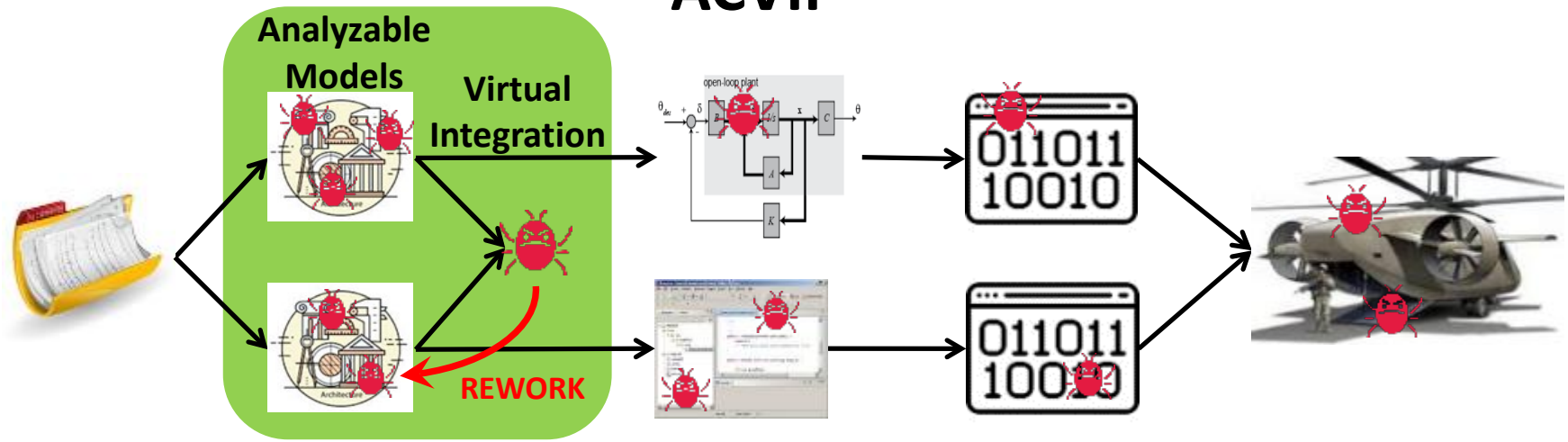
What is ACVIP?

The Architecture Centric Virtual Integration Process (ACVIP) consists of process, methods, and tools to perform model-based virtual integration and analysis of the embedded computing systems.

Traditional Development Process



ACVIP



A Project Gone Wrong

A video of a newscast will be shown at this point.

Common Project Issues

Exercise - PM Issues:
In this exercise we brainstorm a number of issues that can befall projects from a program management perspective. These could come from the example scenario or from their own experience. These issues may come in at different levels of detail.

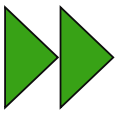
See the Training workbook tab “List PM Issues”.

Project Management Goals

List project management goals that ACVIP might support on a complex embedded software system development project.

Project Management Goals

1. Reduce risk of project cost and schedule overrun
2. Reduce project technical risk
3. Achieve Modular Open Systems Approach (MOSA) goals
4. Increase affordable capability
5. Streamline accreditation and verification
6. Support subsequent lifecycle upgrades
7. Others



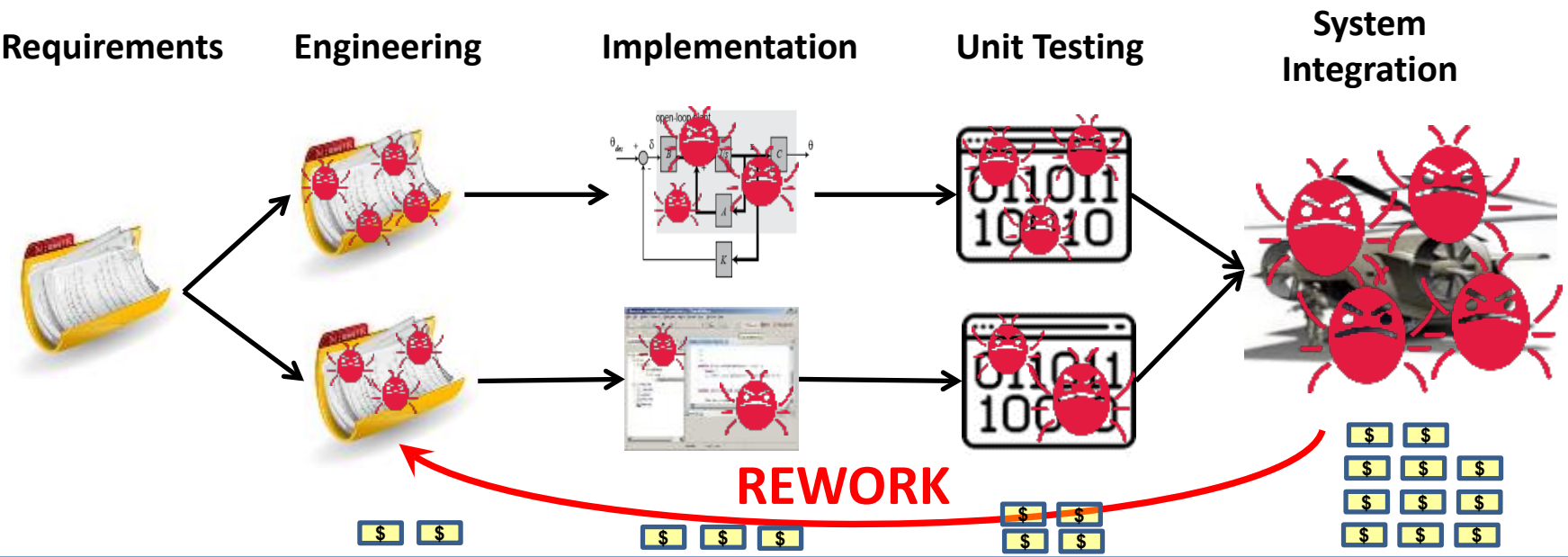
Reduce Risk of Project Overrun

Project cost and schedule risk management begins with risk identification

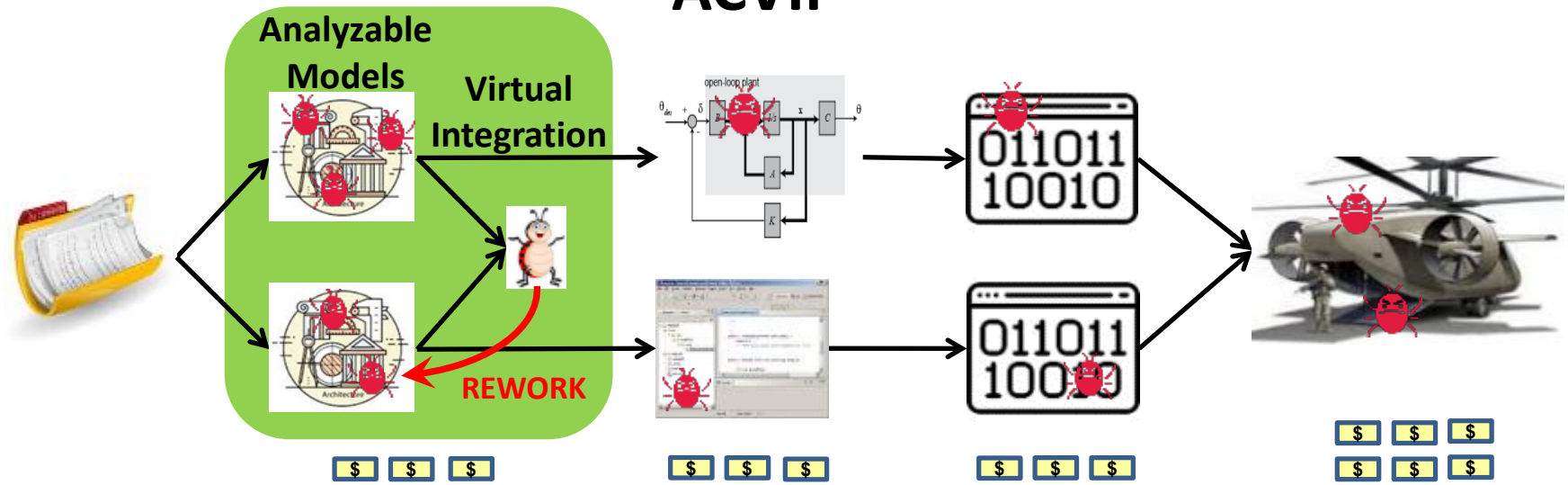
1. Detect early on, defects in embedded system architectures through early virtual integration and analysis of models
2. Address issues early when there is less impact on cost and schedule
3. Support parallel development

Meeting cost and schedule are key goals for program management

Traditional Development Process



ACVIP



Costs to Find and Correct Defects

Phase	Requirements	Design	Code	Test	Integration
Defects are introduced	35%	35%	20%	7%	3%
Defects are found	1%	3%	16%	46%	34%
Costs to correct	.03%	.2%	2%	36%	62%

Based on data from: "ROI Analysis of the System Architecture Virtual Integration Initiative", SEI, 2018

Finding and addressing defects early saves cost

Adventium LABS

BACK TO PROJECT MANAGEMENT GOALS

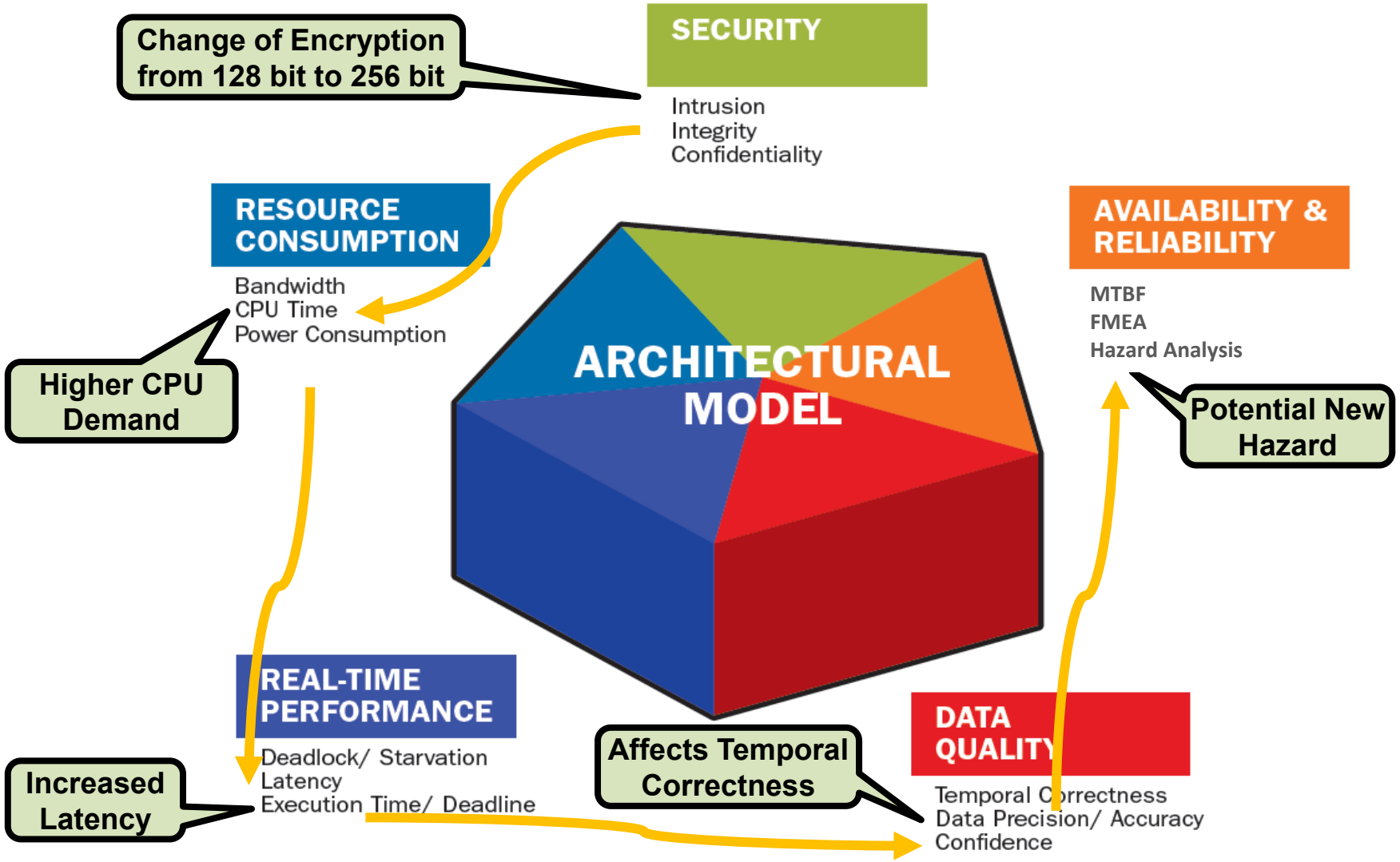
Reduce Project Technical Risk

Technical risk management is an important goal for projects using new technologies or challenging performance requirements.

1. Reduce ambiguity and documentation errors
2. Improve consistency and interoperability between multiple developers
3. Retain / leverage institutional knowledge
4. Support analysis at all stages of development

Uncertainties contribute to program technical risks

Hidden Dependencies Can Increase Technical Risk



Adapted from: Presentation: AADL Overview and Analysis Methods by Bruce Lewis and Jerome Hugues, July 2008.

Small changes can have system wide ramifications

Adventium LABS

BACK TO PROJECT MANAGEMENT GOALS

Adventium LABS

A Modular Open Systems Approach (MOSA) is required for major defense acquisition programs per National Defense Authorization Act of 2017.

1. Modular design
2. Major system components severable to support component addition, removal, or replacement
3. Key interfaces defined between major system components
4. Interfaces comply with industry standards

MOSA provides opportunities for enhanced competition

Adventium LABS

BACK TO PROJECT MANAGEMENT GOALS

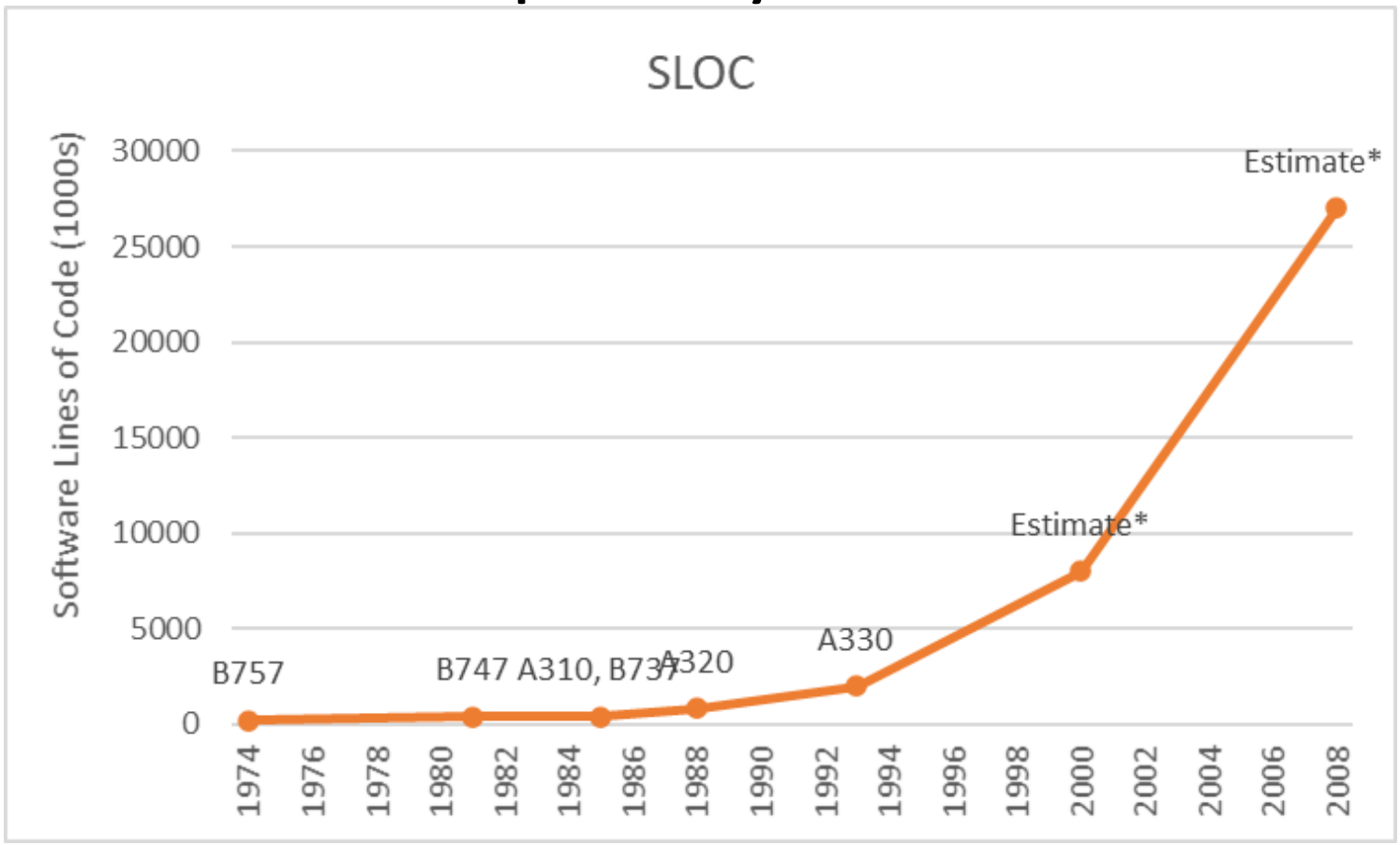
Increase Affordable Capability

Field better capabilities more rapidly.

1. Improve system quality
2. Increase affordable capability by using trade studies between cost versus capabilities
3. Trade studies allow a try-before-you-buy approach.
4. When issues arise, capabilities are often sacrificed in order to stay within budget and schedule

Trade studies support more efficient program resource allocation

Increased Capability = More Software



*Estimate based on doubling of SLOC every 4 years. Based on data from: "ROI Analysis of the System Architecture Virtual Integration Initiative", SEI, 2018

Reducing risk and improving efficiency enables added capability

Adventium LABS

BACK TO PROJECT MANAGEMENT GOALS

Streamline Certifications

Reduce the costs and increase the positive outcomes of certification, verification and approval processes.

1. Follow required certification processes
2. Conduct certification analysis throughout development to reduce surprises late in the project
3. Produce reports in required formats

Analysis results serve as evidence for certification and approval

Adventium LABS

BACK TO PROJECT MANAGEMENT GOALS

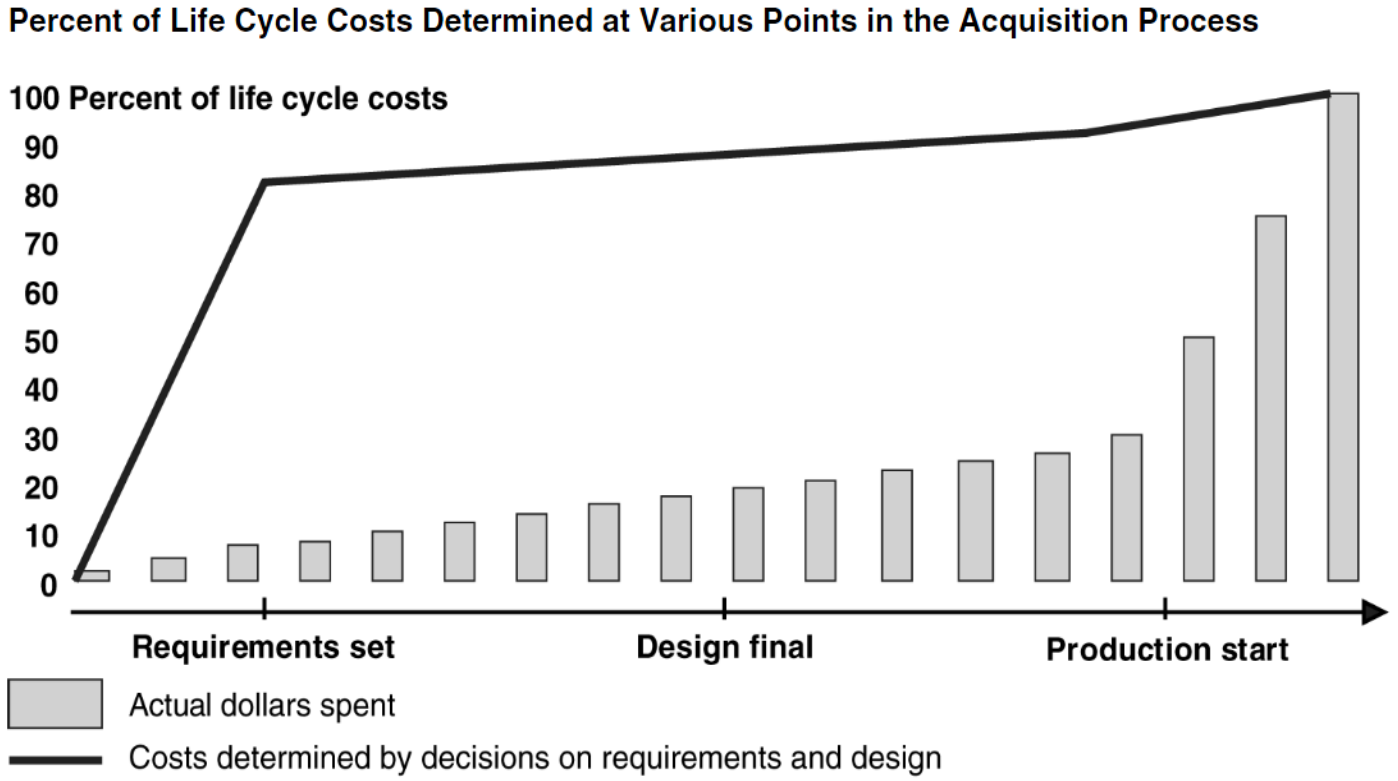
Aid Future System Development and Upgrades

Consider the life cycle costs and management of systems across the full program

1. Consider costs for operation and sustainment early
2. Anticipate and aid future system upgrades
3. Provide assets that reduce cost, schedule, and risk for future upgrades and programs
4. The cost of the creation of some assets may be spread across product lines or families of systems

Artifacts such as models, analysis results, and reports can be shared across projects

Life Cycle Costs



Operations & Sustainment Costs

- 70% of life cycle cost is O&S
- 90% of O&S cost has been built-in by end of requirements engineering.

From:2003 study from Defense Acquisition University, "Setting Requirements Differently Could Reduce Weapon Systems' Total Ownership Costs"

Most costs are determined early but paid late

Adventium LABS

BACK TO PROJECT MANAGEMENT GOALS

Model Based Project Goals

Adventium LABS

BACK TO PROJECT MANAGEMENT GOALS

ACVIP Solutions

- Single common model for multiple uses
- Common interface definitions
- Utilizes industry standards
- Analysis tools for requirements verification/validation
- Iterative model refinement
- Virtual integration
- Automated continuous integration and analysis
- Customizable report generation

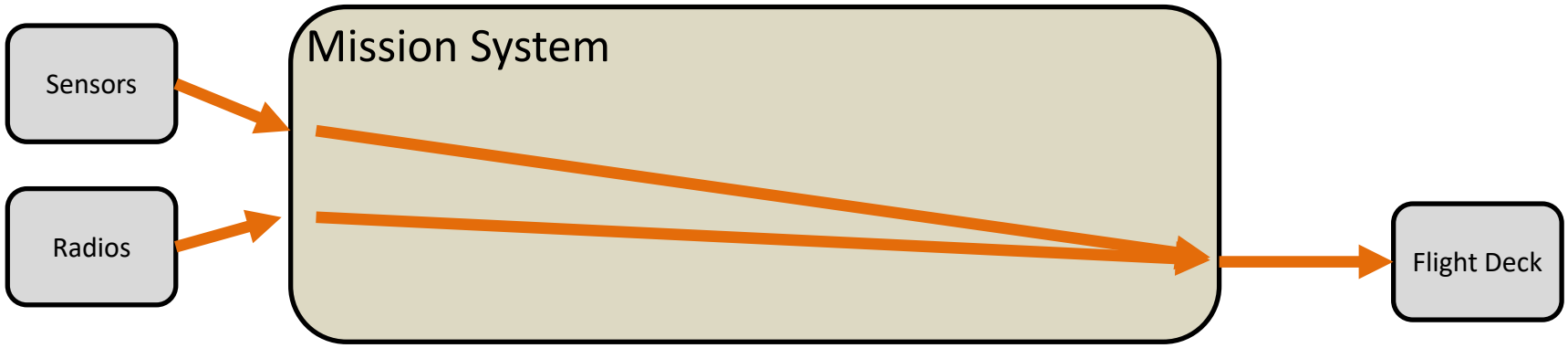
Model Based

A model is a representation of one or more concepts that may be realized in the physical world

- Models should conform to industry standards
  - Interoperability
- Multiple models can be integrated to form a larger system
  - Supports parallel development by multiple organizations
- The ACVIP model serves multiple purposes:
  - Analysis
  - Documentation
  - Communication
- ACVIP Models can have varying levels of detail or fidelity

ACVIP employs models for integration and analysis

Iterative Model Refinement

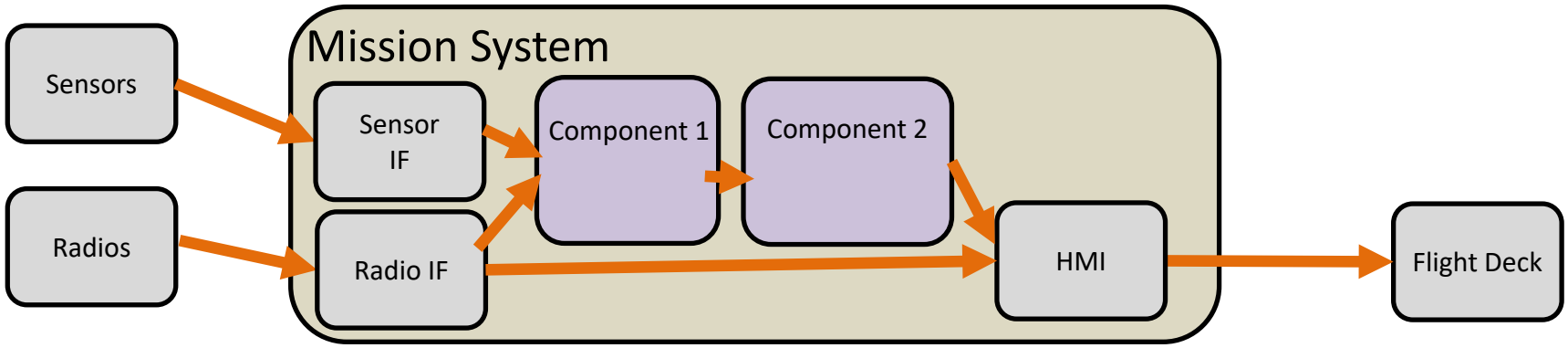


Low model fidelity analysis:

- MILS
- Safety / Risk
- Assign Latency Budgets

ACVIP analysis starts early in the design process

Iterative Model Refinement

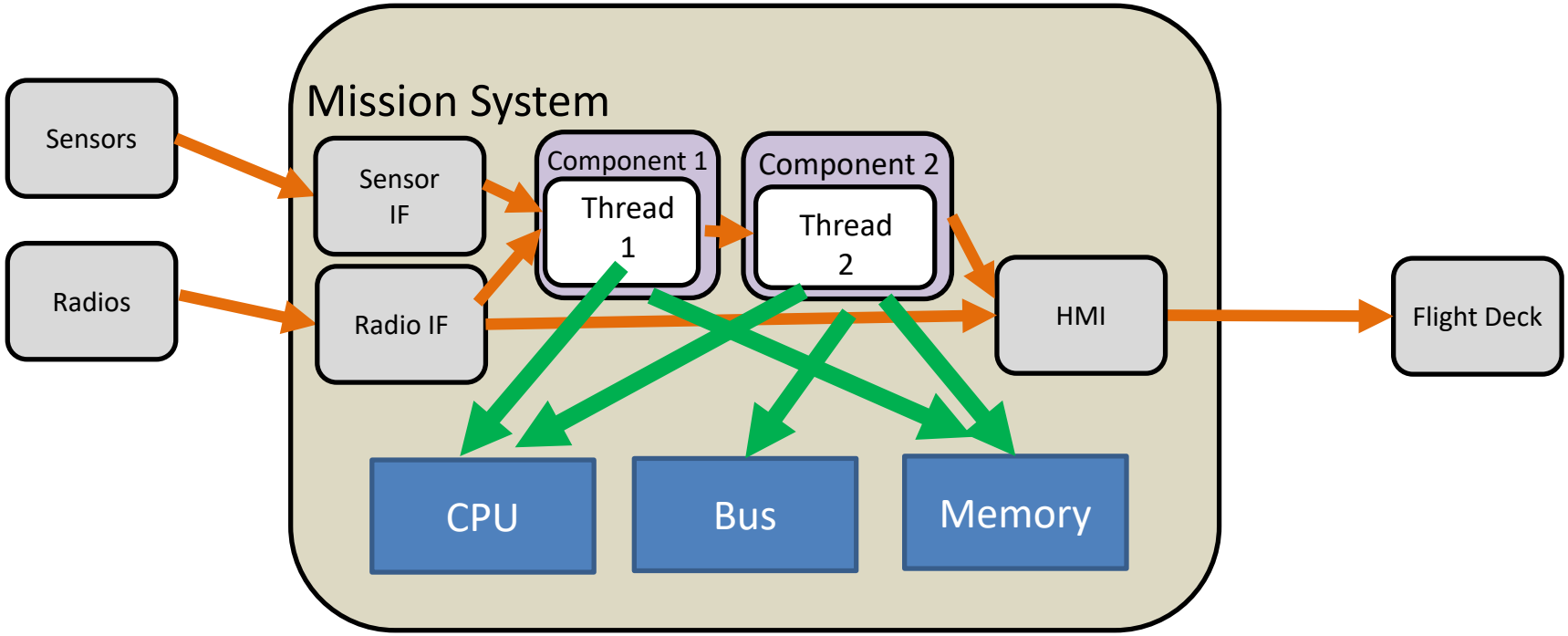


Mid fidelity analysis:

- Inter-connection consistency
- Error handling and propagation
- Latency budgets by component

Additional model fidelity leads to additional analysis detail

Iterative Model Refinement



High fidelity analysis:

- CPU, Memory and Bus Utilization
- Latency

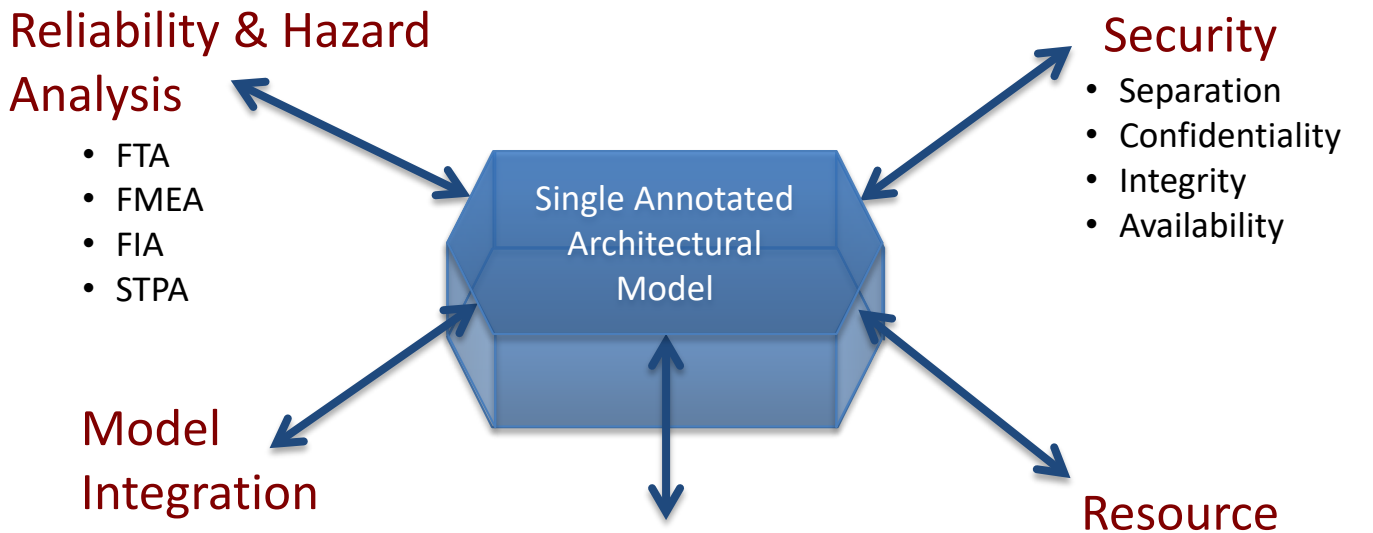
Detailed models provide detailed analysis results including code generation

Model Fidelity (level of detail)

Fidelity Level	Example Model Elements
1	System, Inputs and Outputs, Flows
2	SubComponents, Connections, Processes, EMV2 Error States and Transitions
3	Threads and Thread Groups, End to End Flows, Modes, EMV2 Error Propagations
4	Processor, Memory and Bus Hardware Components, Bindings
5	Behaviors, Subprogram Calls

Different portions of a model may be at different fidelity levels

ACVIP Analysis Tools



Reliability & Hazard Analysis

- FTA
- FMEA
- FIA
- STPA

Security

- Separation
- Confidentiality
- Integrity
- Availability

Model Integration Quality

- Data interface compatibility
- Data interface consistency
- Standard Conformance

Real-time Performance

- Execution time/Deadline
- Deadlock/starvation
- Latency
- Schedulability

Resource Consumption

- Bandwidth
- CPU time
- Power consumption
- Weight

Common model supports a range of analyses

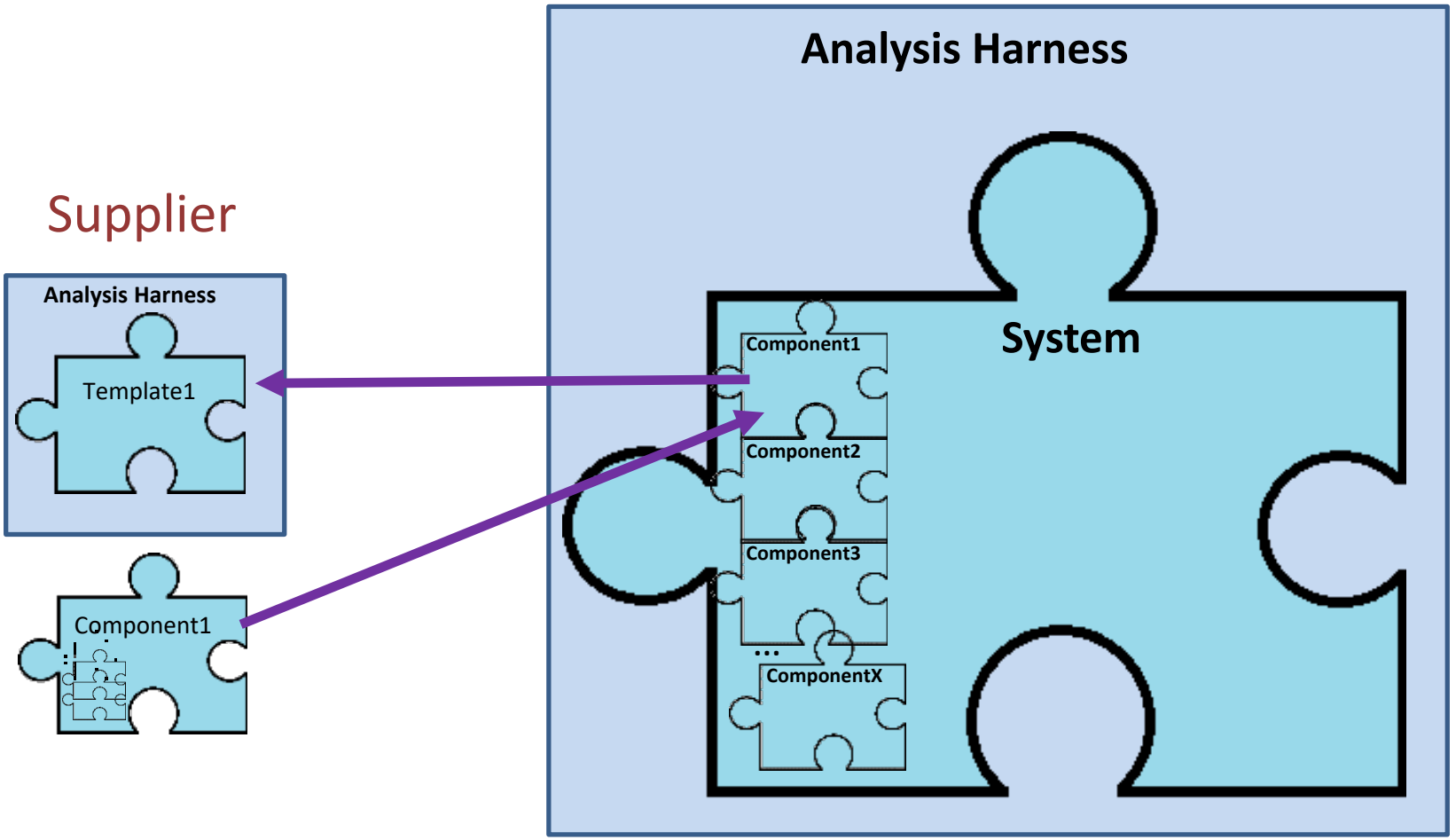
Virtual Integration

- Virtual Integration combines model components into a larger system to support testing and analysis
- ACVIP Virtual Integration:
  - Can find interface incompatibilities
  - Supports mixed fidelity models
  - Supports hierarchically organized models
  - Can be performed early in a design process
  - Should be repeated whenever a model changes
  - Can be applied at multiple customer / supplier levels
  - Can be automated

Virtual integration should be performed early and often

Virtual Integration & Analysis

Customer



Component models integrate into system models

ACVIP Solutions

- Single common model for multiple uses
- Common interface definitions
- Utilizes industry standards
- Analysis tools for requirements verification
- Iterative model refinement
- Virtual integration
- Automated continuous integration and analysis
- Customizable report generation

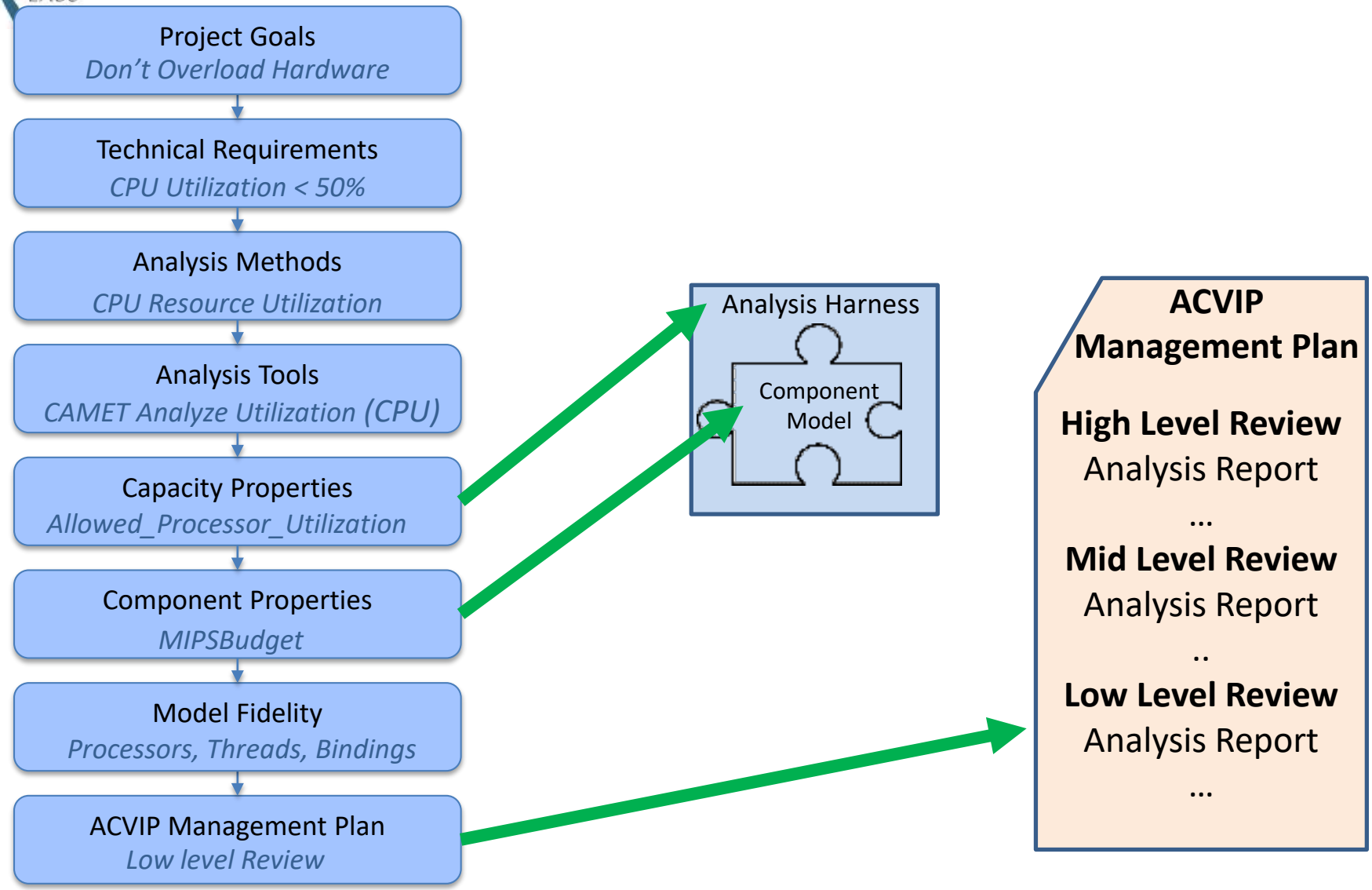
ACVIP Solutions for Issues

Exercise – ACVIP Features:

In this exercise ACVIP features are matched up with the issues solicited earlier.

Using the Training workbook “Match Issues with Solutions”.

ACVIP Workflow Overview



From goals and requirements to properties and analysis

ACVIP Training Outline

1.0 ACVIP Overview
- Common Project Issues
- Project Goals and ACVIP Solutions

2.0 Structure Models for Delivery and Virtual Integration
- AADL for Modeling Embedded Systems
- Modeling Guidelines

3.0 ACVIP Analysis Tools: Risk, Safety and Security
- Risk and Safety Analysis: SESSAF, FTA, FIA
- Security Analysis: MILS, MADS, RMF

4.0 ACVIP Analysis Tools: Utilization, Timing, and Behavior
- Utilization and Timing Analysis: OSATE & CAMET Tools
- Behavior Analysis: CAMET SLICED Tool

5.0 Model Based Development
- Multi-Organizational Model Development Exercise

6.0 Model Based Integration
- Multi-Organizational Virtual Integration Exercise

7.0 Assure Model Conformance & Certification
- Integrating Models from Other Sources
- Managing IP

8.0 Develop ACVIP Management Plan
- Common Project Issues
- Project Goals and ACVIP Solutions

ACVIP Modeling & Analysis Training

An interactive hands-on training course to give participants an understanding of the **Architecture Centric Virtual Integration Process (ACVIP)**, and to provide hands-on experience with key modeling techniques and analysis tools.

ACVIP includes modeling methods, processes, and analysis tools that can be applied to embedded computing system design projects to:

- Reduce technical risk
- Reduce cost and schedule overruns
- Support conformance to industry standards & required certifications

Course Details

- Interactive, hands-on OSATE and AADL modeling and analysis
- Live, virtual class via Teams
- Based on the ACVIP Modeling and Analysis Handbook
- Target audience: System designers and developers
- 1st session (overview) will be useful for project managers
- Experience with AADL useful but not required
- 8 sessions lasting about 60 - 90 min each
- Classes begin March 2021

Course Outline

1. Benefits of ACVIP
2. Developing an ACVIP Management Plan
3. Structuring Models for Delivery and Virtual Integration
4. Defining the Model Content Required for Analysis
5. Incremental Design, Analysis, and Resolution
6. Model Based Development
7. Model Based Integration
8. Assuring Model Conformance & Certification