Pursuant to ITAR 120.10, this document contains no technical data

Raytheon

Customer Success Is Our Mission

Engineering Practices and Patterns for Rapid BIT Evolution

James Brewer
Principal Systems Engineer
Raytheon Missile Systems

October 2011

NDIA
National Defense Industrial Association

Intent of Presentation

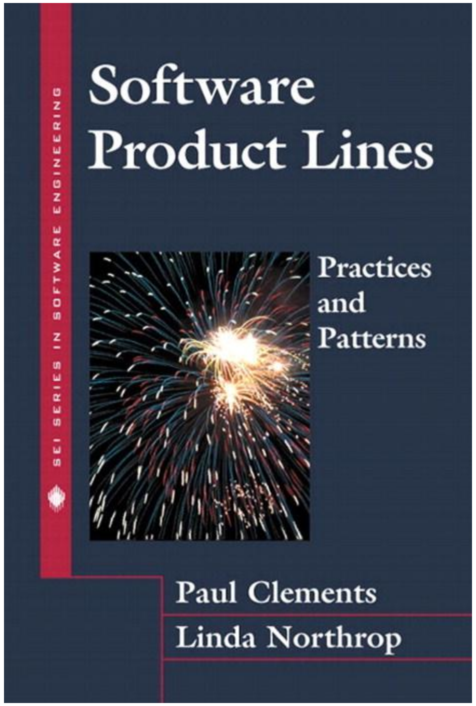
- Propose methods for rapid Built-In Test (BIT) evolution
- Present ideas in nonproprietary terms
- Suggest a product line approach
- Optimize product self-test capability
- Reduce overall life cycle test costs
- Maximize tactical verification of system availability

Evolution vs. Development

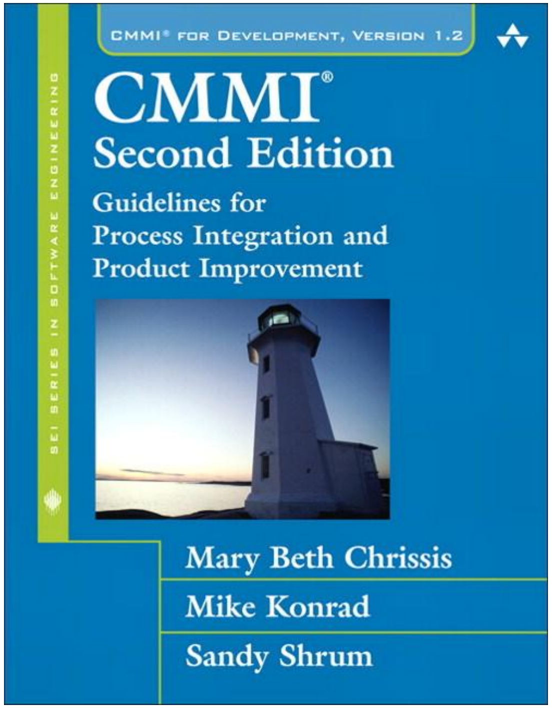
- It depends on where you’re at
  - New system = Development
  - Design spin = Evolution
- Future favors more incremental evolution of systems
- Spiral development and rollout of capabilities
- Warfighter demand for schedule
- Budget demand for affordability
- Methods in this presentation scalable to life cycle phase

References

- Product Line Practices and Patterns definitions as well as Capability Maturity Model Integrated (CMMI) Process Area mappings are taken from the following Carnegie Mellon University texts, referenced here by permission of the Software Engineering Institute (SEI) and the text book publishers.



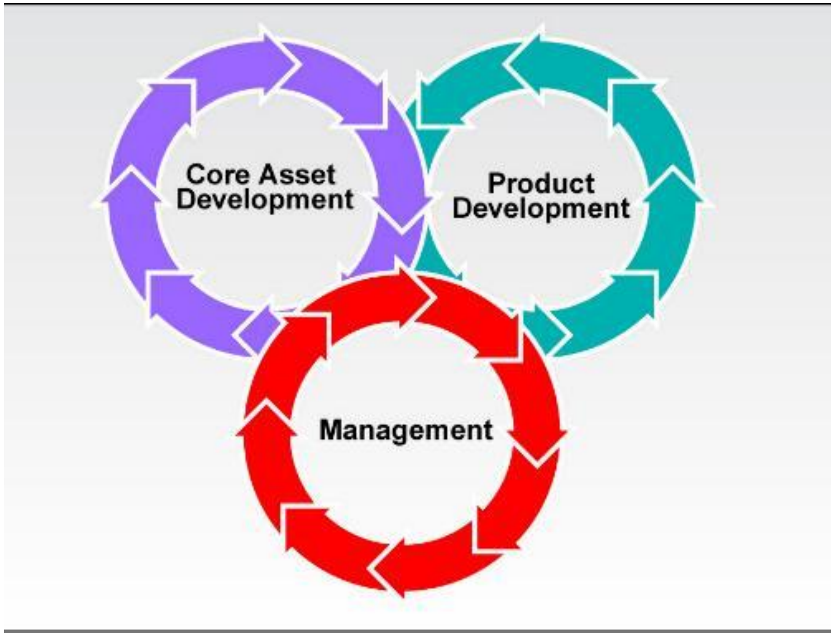
© 2002 Addison-Wesley


CMMI® FOR DEVELOPMENT, VERSION 1.2
SEI SERIES IN SOFTWARE ENGINEERING
CMMI®
Second Edition
Guidelines for Process Integration and Product Improvement
Mary Beth Chrissis
Mike Konrad
Sandy Shrum

© 2007 Pearson Education, Inc.

The SEI Product Line Framework

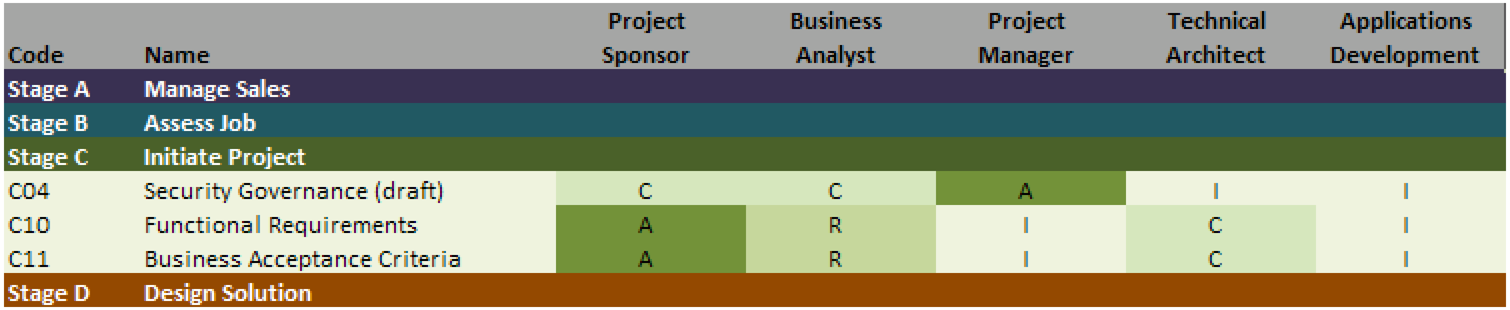
- Three Essential Activities
  - Core Asset Development
  - Product Development
  - Management
- Practice Areas
  - Engineering
  - Technical Management
  - Organizational Management
- Practice Patterns
  - Easy to understand for practical application
  - Context-Problem-Solution Schema



© 2002 Software Engineering Institute, Carnegie Mellon University
http://www.sei.cmu.edu/productlines/frame_report/PL.essential.act.htm

RASCI Responsibility Matrix

- Responsible
- Accountable (Approver or Authority)
- Supporting
- Consulted
- Informed

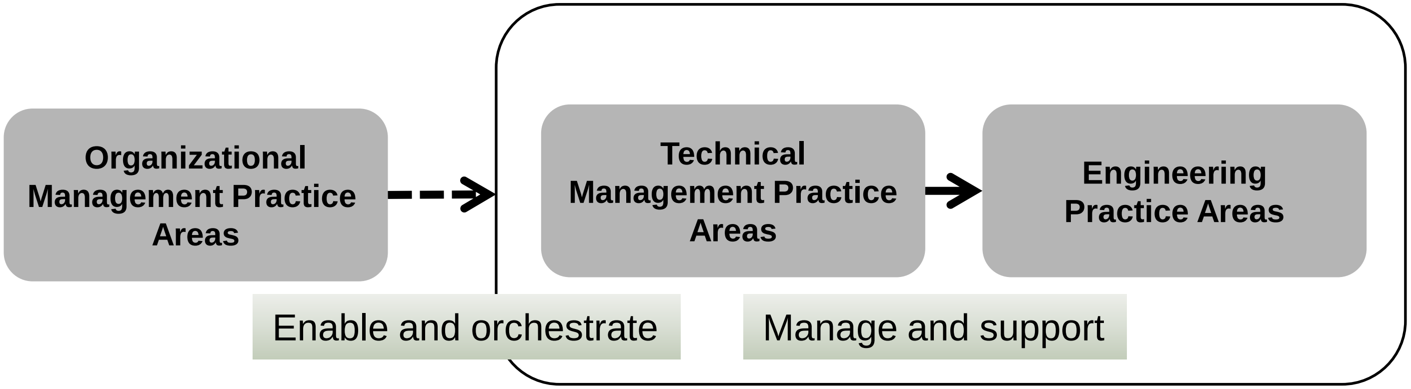


Code	Name	Project Sponsor	Business Analyst	Project Manager	Technical Architect	Applications Development
Stage A	Manage Sales					
Stage B	Assess Job					
Stage C	Initiate Project					
C04	Security Governance (draft)	C	C	A	I	I
C10	Functional Requirements	A	R	I	C	I
C11	Business Acceptance Criteria	A	R	I	C	I
Stage D	Design Solution					

Wikipedia, *Responsibility assignment matrix,* last modified 15 August 2011
http://en.wikipedia.org/wiki/Responsibility_assignment_matrix

RASCI Responsibility Matrix

- "Curriculum" Pattern
- Groups all 29 product line practice areas into three categories based on skill sets


Organizational Management Practice Areas
Technical Management Practice Areas
Engineering Practice Areas
Enable and orchestrate
Manage and support

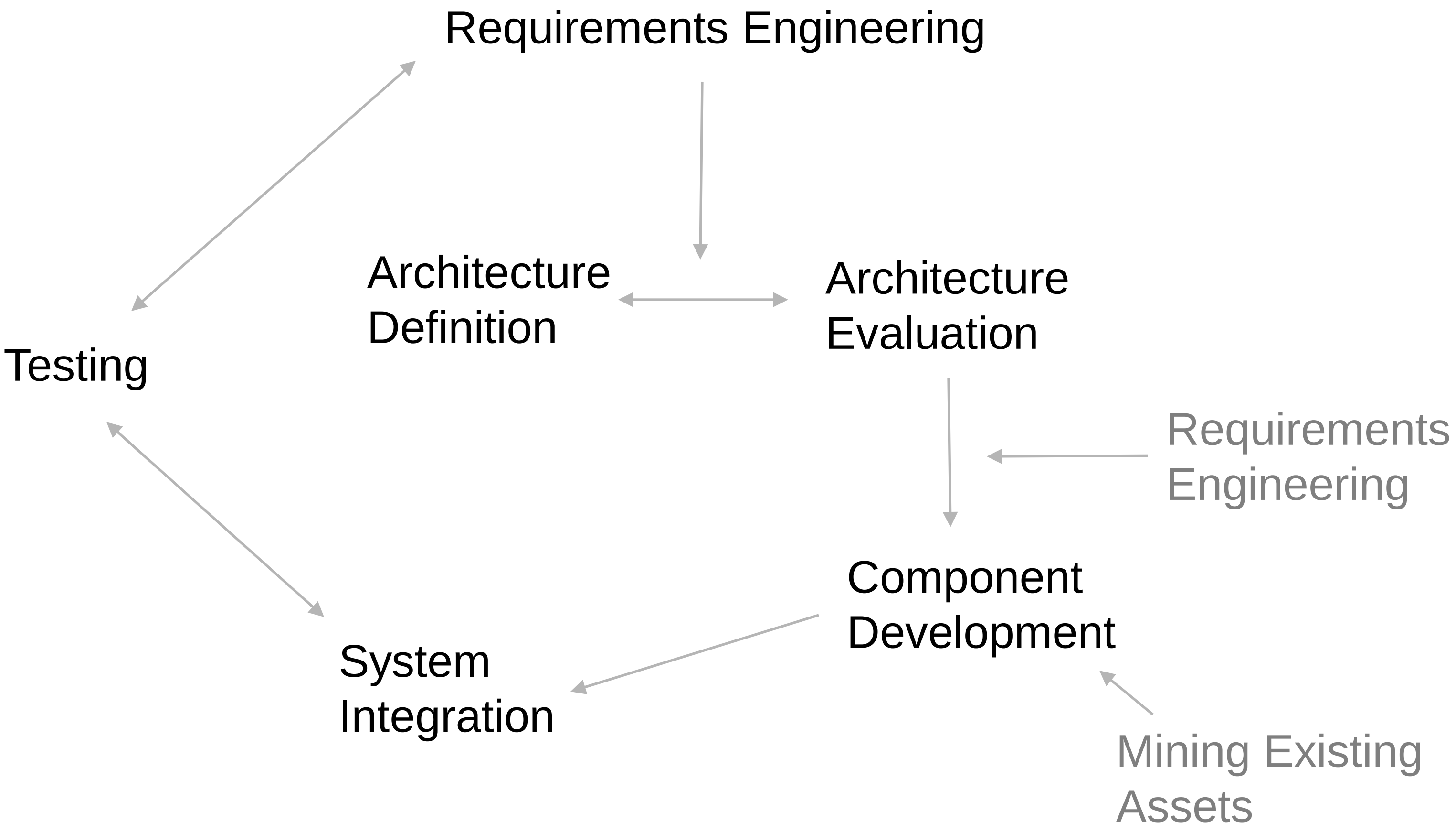
© 2002 Software Engineering Institute, Carnegie Mellon University
http://www.sei.cmu.edu/productlines/frame_report/productLPAs.htm

Engineering Practice Areas

	CMMI Process Areas
▪ Requirements Engineering *	Requirements Development
▪ Architecture Definition *	
▪ Architecture Evaluation *	Technical Solution
▪ Mining Existing Assets *	
▪ COTS Utilization	
▪ Component Development	
▪ System Integration	
▪ Testing *	Verification & Validation
▪ Understanding Relevant Domains	

BIT Development Process Cycle

- “Product Builder” Pattern



Requirements Engineering

Architecture Definition

Architecture Evaluation

Testing

Requirements Engineering

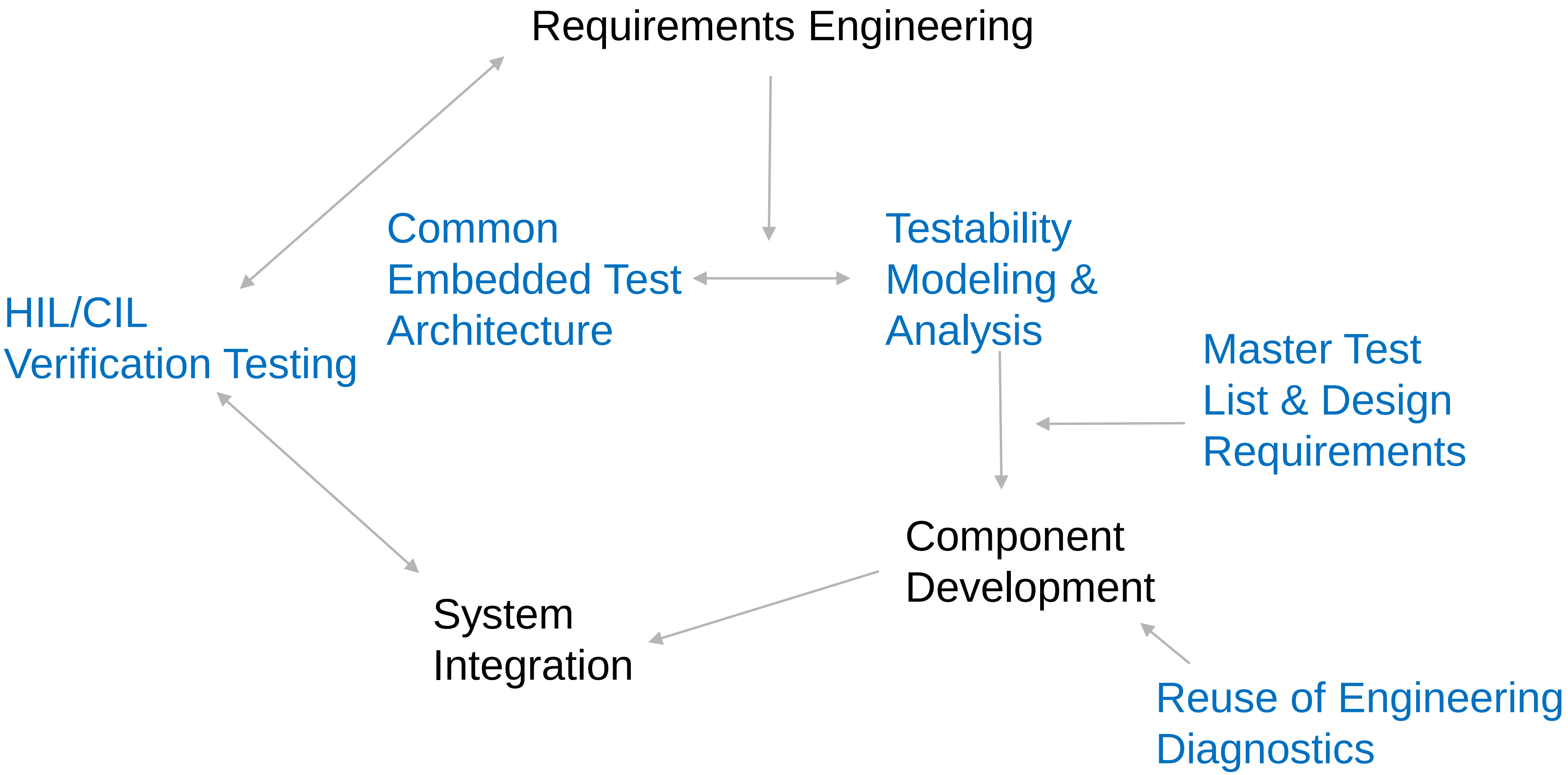
Component Development

System Integration

Mining Existing Assets

BIT Development Process Cycle

- “Product Builder” Pattern



Requirements Engineering

Common Embedded Test Architecture

Testability Modeling & Analysis

HIL/CIL Verification Testing

Master Test List & Design Requirements

Component Development

System Integration

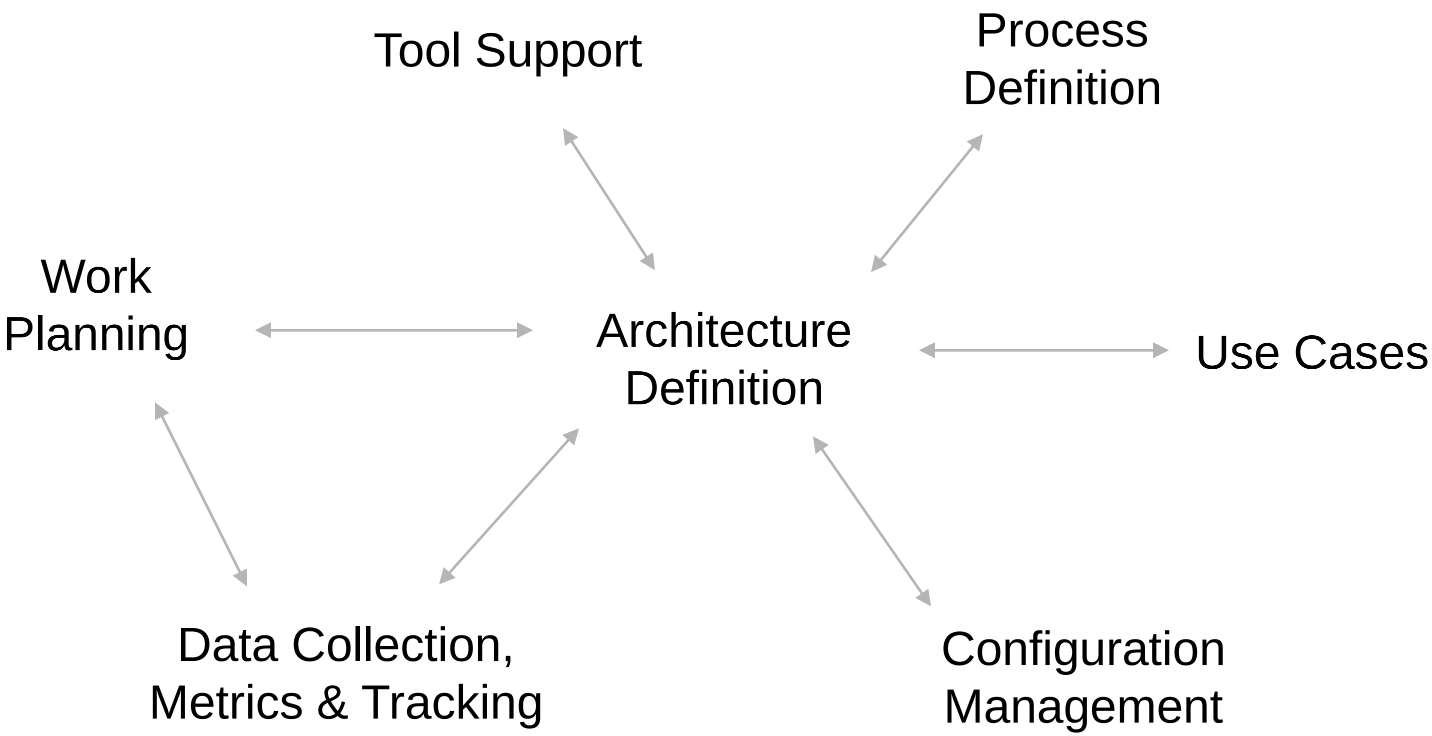
Reuse of Engineering Diagnostics

Common Embedded Test Architecture

- “Each Asset” Pattern
  - Consists of practice areas that should be used whenever any asset in the core asset base is being developed
- Context:
  - The asset to be developed is known, the asset has specifications and who will complete the task has been determined
- Problem:
  - To use the proper set of practices to develop the asset so that it will be an effective member of the product line’s asset base
- Solution:
  - Develop the asset in a way that is appropriate for its type and with the appropriate tools
  - Attach a process and use cases to its definition
  - Have a work plan and track progress
  - Test the asset and put it under configuration management

Common Embedded Test Architecture

- “Each Asset” Pattern



Tool Support

Process Definition

Work Planning

Architecture Definition

Use Cases

Data Collection, Metrics & Tracking

Configuration Management

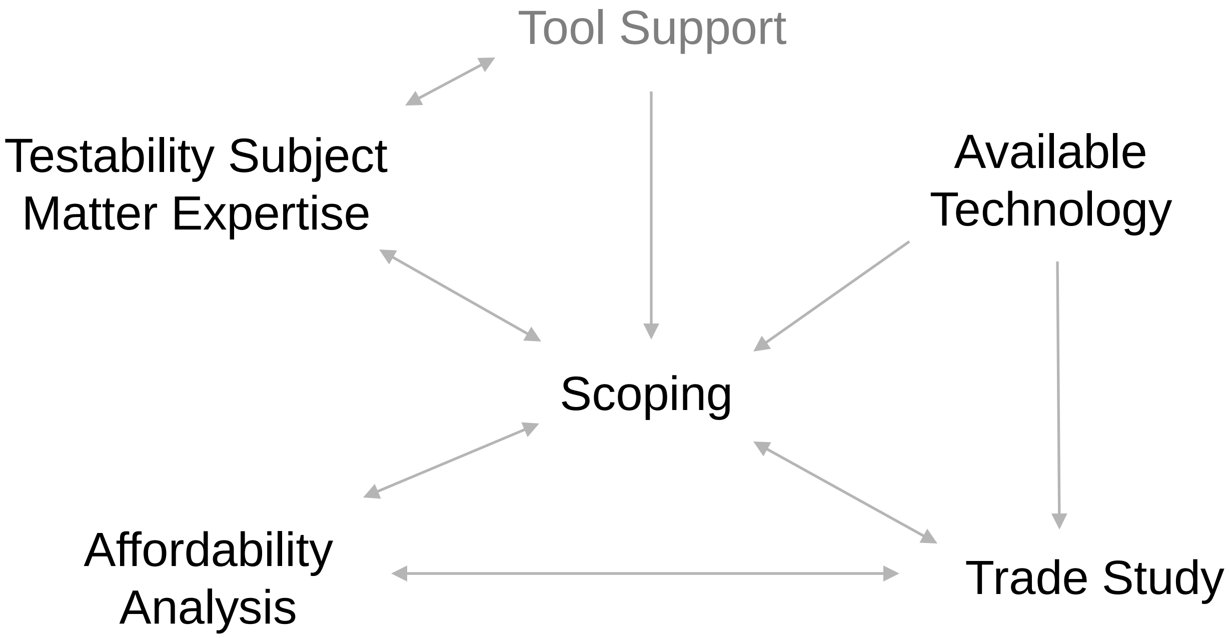
Testability Modeling & Analysis

- “What to Build” Pattern
  - Consists of practice areas that help determine what products, components or design elements should be built
- Context:
  - The organization has decided to field a product and knows its operational requirements
- Problem:
  - To determine what should be included for self-test capability
- Solution:
  - Collect information on product concept of operations
  - Collect information on product current design
  - Assess gaps in functional self-test coverage
  - Collect information on available technology
  - Conduct cost vs. performance trade studies
  - Recommend changes to design and/or requirements

Raytheon

Testability Modeling & Analysis

- “What to Build” Pattern



Tool Support

Testability Subject Matter Expertise

Available Technology

Scoping

Affordability Analysis

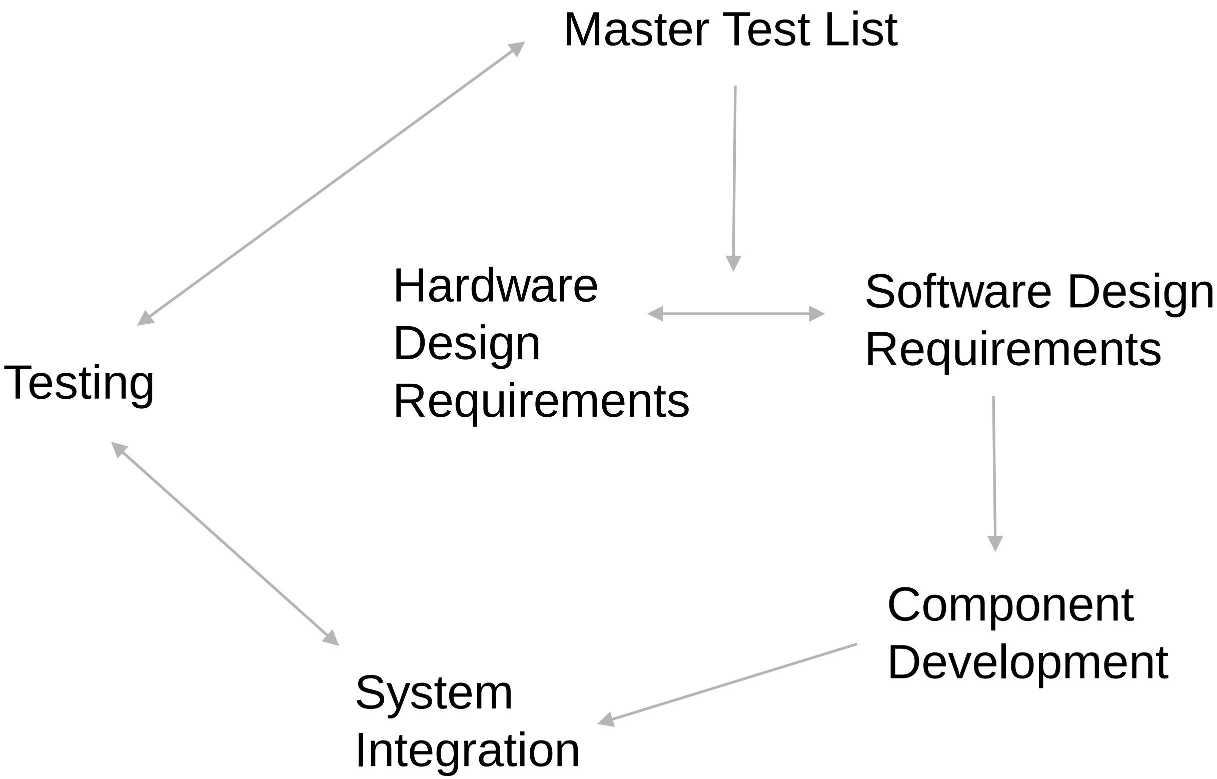
Trade Study

Master Test List & Design Requirements

- “Product Builder” Pattern
  - Consists of practice areas that should be used whenever any product in the product line is being developed
- Context:
  - An organization has already established the production plan, the production capability, the core asset base and has designated knowledgeable individuals or groups to develop a product in the product line
- Problem:
  - To develop a product from the core assets using the production plan
- Solution:
  - Follow the production plan using established capabilities
  - Additional components are developed and integrated into those assembled from the asset base
  - Components are integrated and tested according to the production plan

Master Test List & Design Requirements

- "Product Builder" Pattern



Master Test List

Hardware Design Requirements

Software Design Requirements

Testing

Component Development

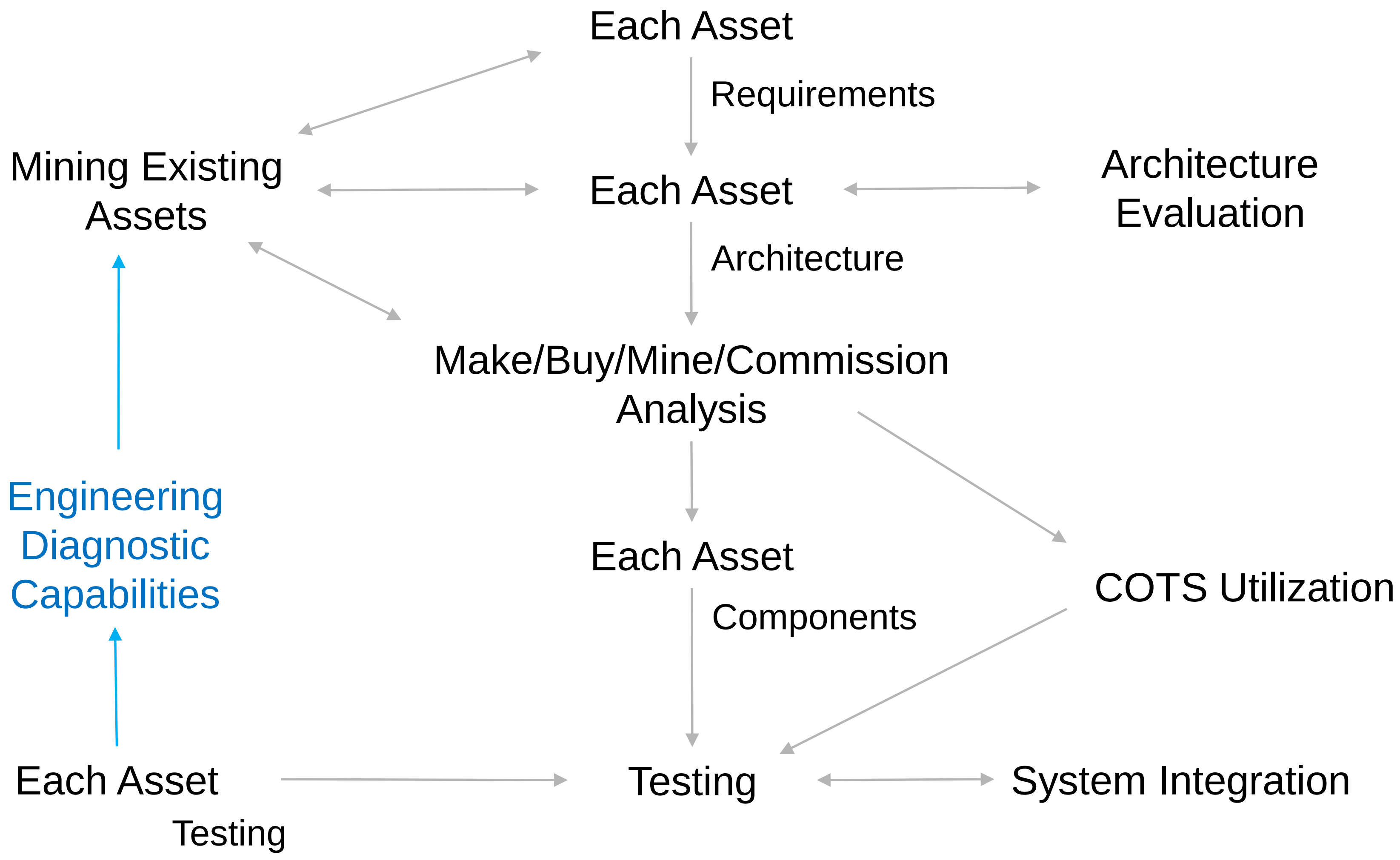
System Integration

Reuse of Engineering Diagnostics

- "Plowed Field" Pattern
  - Consists of practice areas that should be used to develop the product, but will use existing assets in the product scope as much as possible
- Context:
  - An organization knows what product(s) to build and has designated knowledgeable individuals or groups to develop the core assets
- Problem:
  - To develop the core assets that will be joined together to form the new product
- Solution:
  - Identify existing diagnostic capabilities within the current design or methods from engineering developmental testing that could be incorporated into permanent capabilities of the product
  - Define the process by which the reusable components will be used in the construction of the product
  - Determine the best source for each component
  - Test each component before and after product integration

Reuse of Engineering Diagnostics

- "Plowed Field" Pattern



Each Asset

Requirements

Mining Existing Assets

Each Asset

Architecture Evaluation

Architecture

Make/Buy/Mine/Commission Analysis

Engineering Diagnostic Capabilities

Each Asset

Components

COTS Utilization

Each Asset Testing

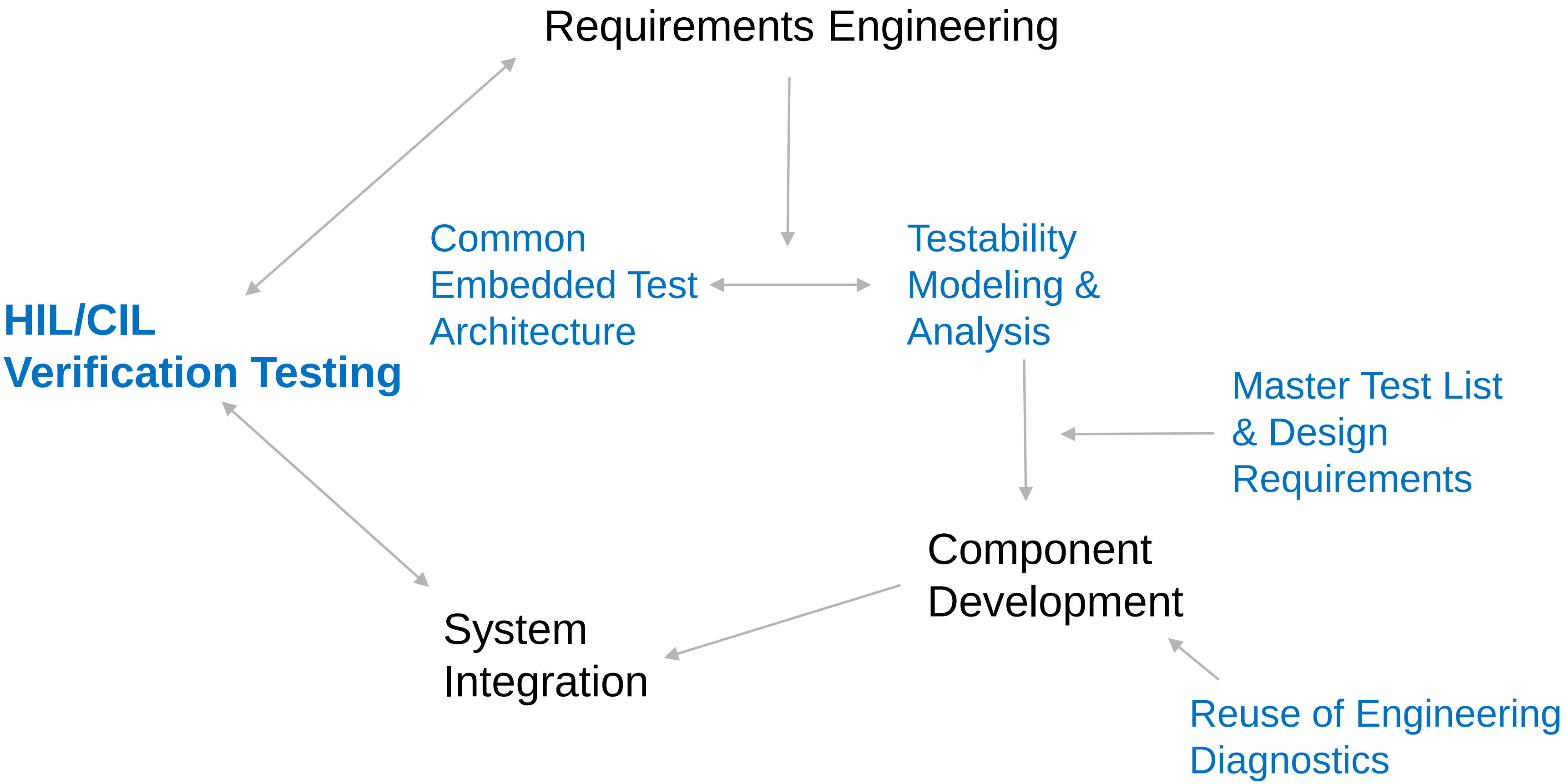
Testing

System Integration

HIL/CIL for BIT Verification Testing

- Hardware-in-the-Loop (HIL), Computer-in-the-Loop (CIL)
- "Product Builder" with "Plowed Field" Pattern
  - Consists of practice areas that should be used whenever any product in the product line is being developed and will use existing assets in the product scope as much as possible.
- Context:
  - An organization has already established a HIL/CIL development plan, knows what product(s) to build and has designated knowledgeable individuals or groups to develop a HIL/CIL BIT Verification capability for the product line.
- Problem:
  - To develop the core assets that will be joined together to form the new HIL/CIL
- Solution:
  - Identify core assets for requirements, architecture, components and their test-related artifacts that will be included in the HIL/CIL testing of the product
  - Define the process by which components will be used in the construction of the HIL/CIL and in the product's BIT verification
  - Determine the best source for each component
  - Test each component before and after system integration

BIT Development Cycle



Requirements Engineering

Common Embedded Test Architecture

Testability Modeling & Analysis

HIL/CIL Verification Testing

Master Test List & Design Requirements

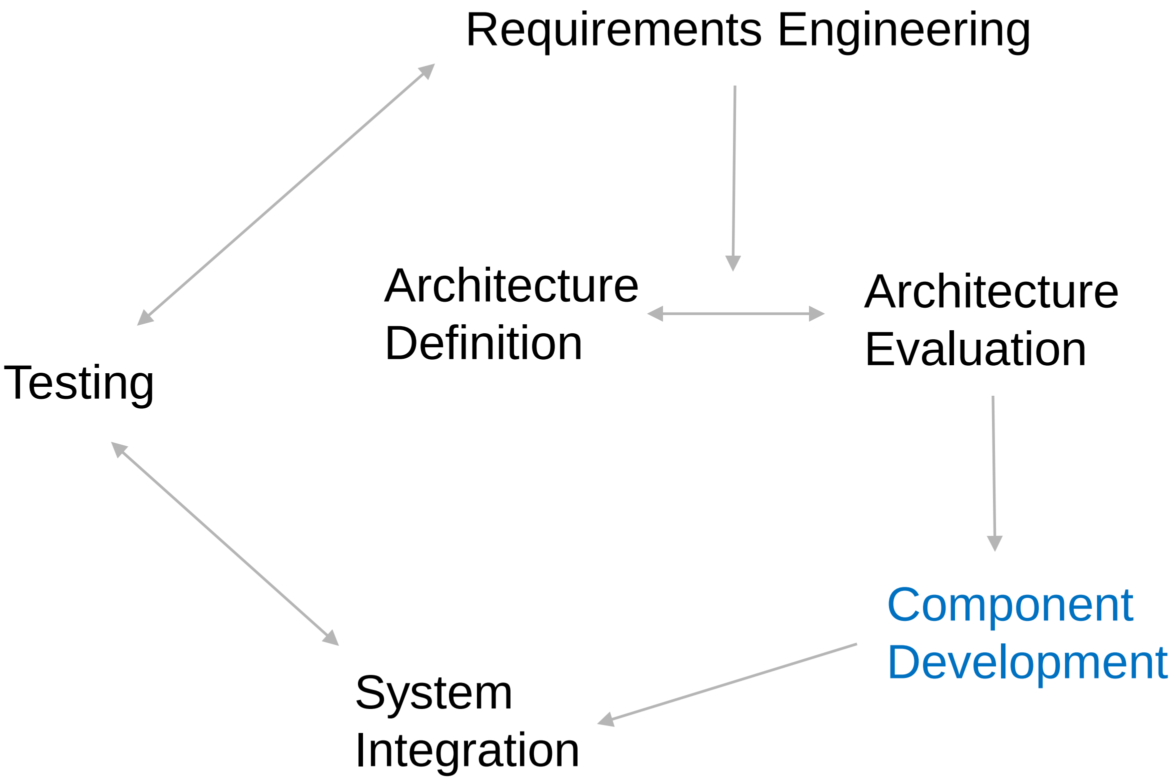
Component Development

System Integration

Reuse of Engineering Diagnostics

HIL/CIL Development Cycle

- “Product Builder” Pattern



Requirements Engineering

Architecture Definition

Architecture Evaluation

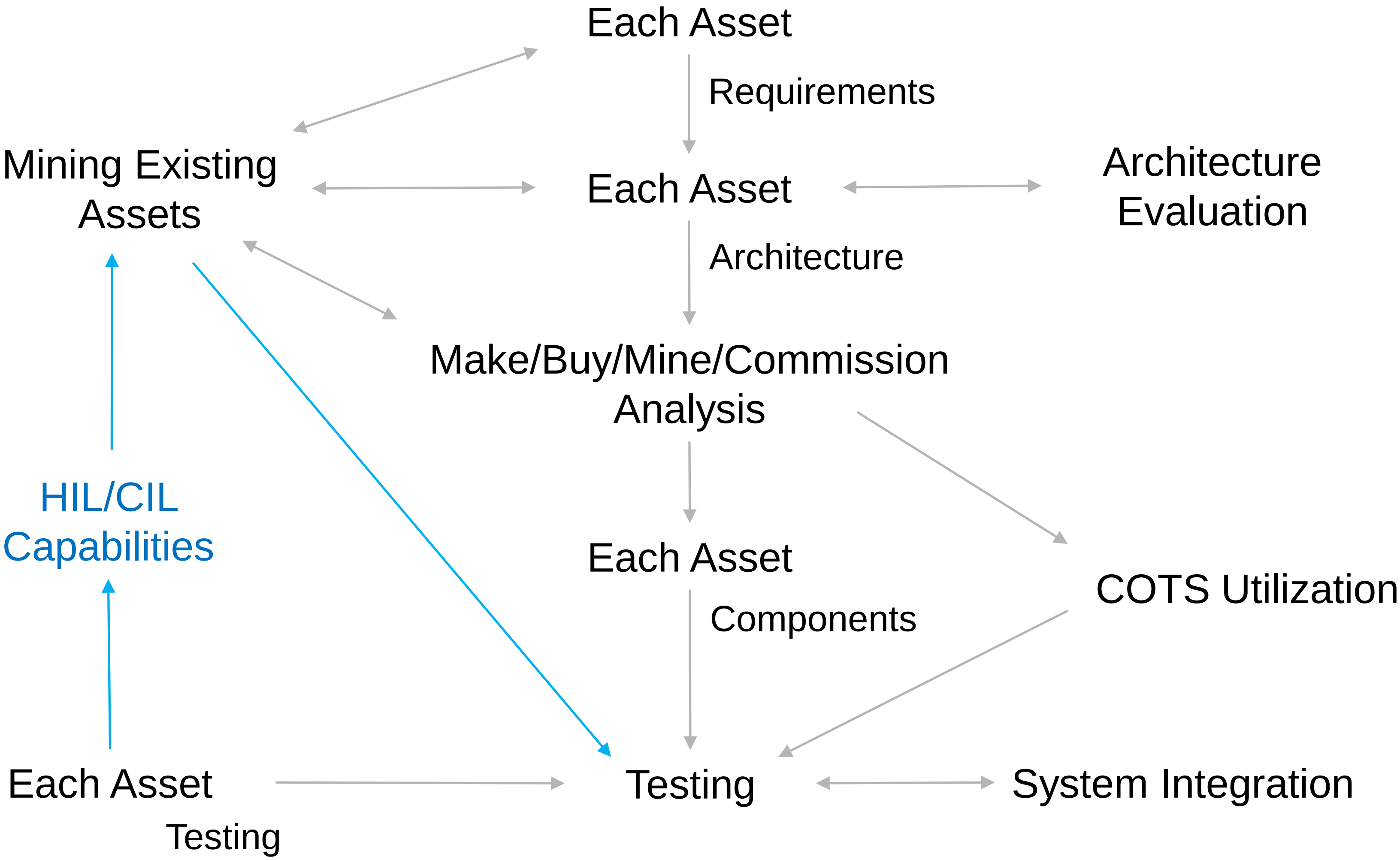
Testing

Component Development

System Integration

HIL/CIL for BIT Verification Testing

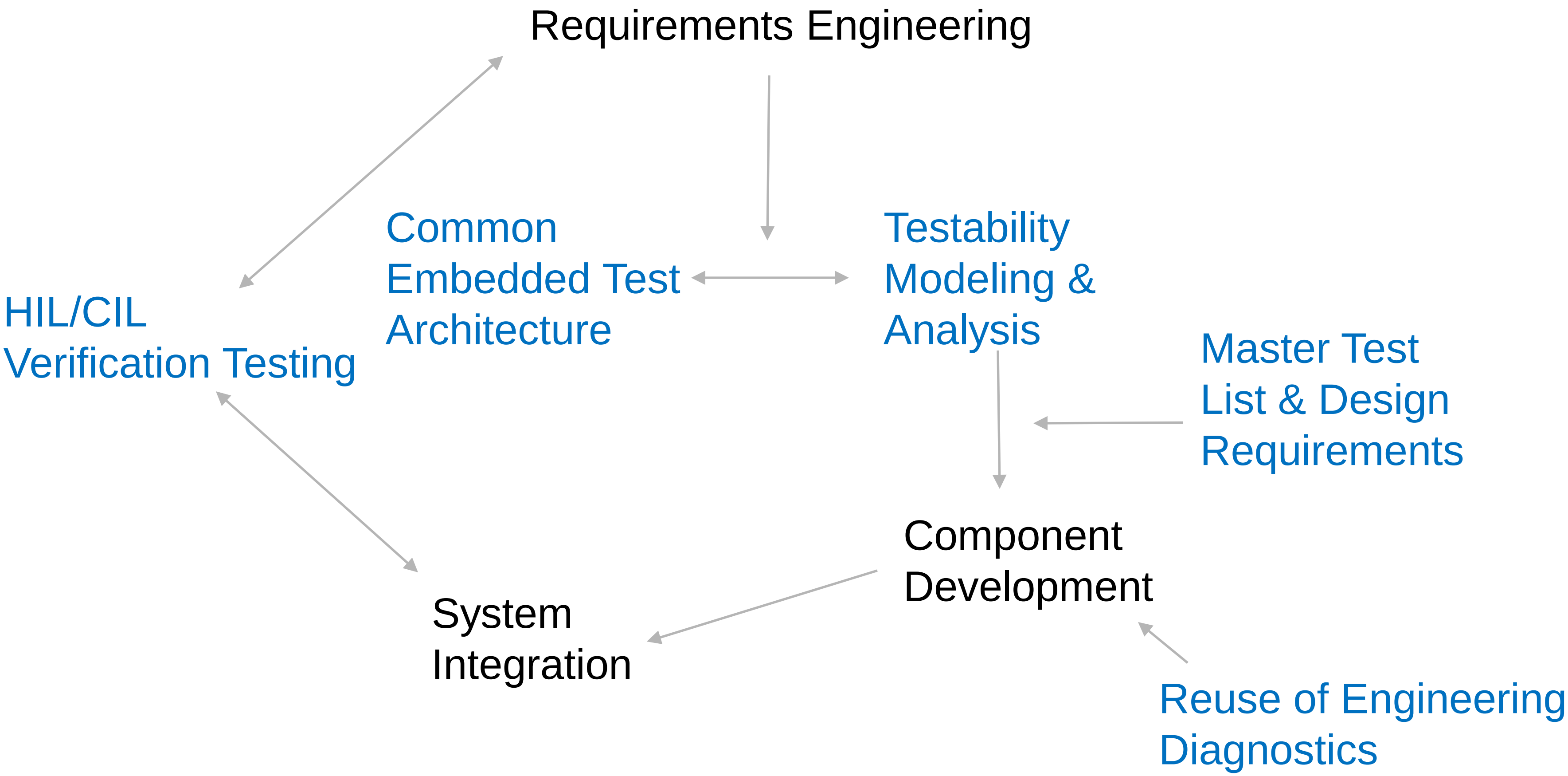
- “Plowed Field” Pattern



Take a breath

Let’s recap

BIT Evolution Process Cycle



Requirements Engineering

Common Embedded Test Architecture

Testability Modeling & Analysis

HIL/CIL Verification Testing

Master Test List & Design Requirements

Component Development

System Integration

Reuse of Engineering Diagnostics

Conclusions

- A product line approach can be adapted to consider the rapid redesign of products for evolutionary BIT capabilities
- The product line approach can be defined in terms of engineering practices and their corresponding practice patterns
- The definitions presented today can be applied to any type of system development within any life cycle stage
- This is because the patterns have been abstracted to apply to any scale of new or existing product design