

**Anatomy of the CMMI
Technical Solution
Process Area**

CMMI Conference 2008
Denver, CO



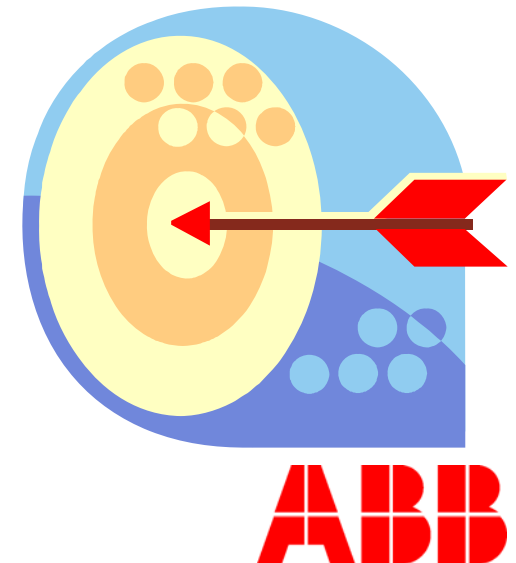
Aldo Dagnino

**ABB Inc.
US Corporate Research
Center
Raleigh, NC**



Presentation Objectives

- Discuss the relationship between the Technical Solution process area and the rest of the Engineering CMMI process areas
- Discuss how the Attribute Driven Design methodology has been used to satisfy the specific goals of the Technical Solution Process Area.

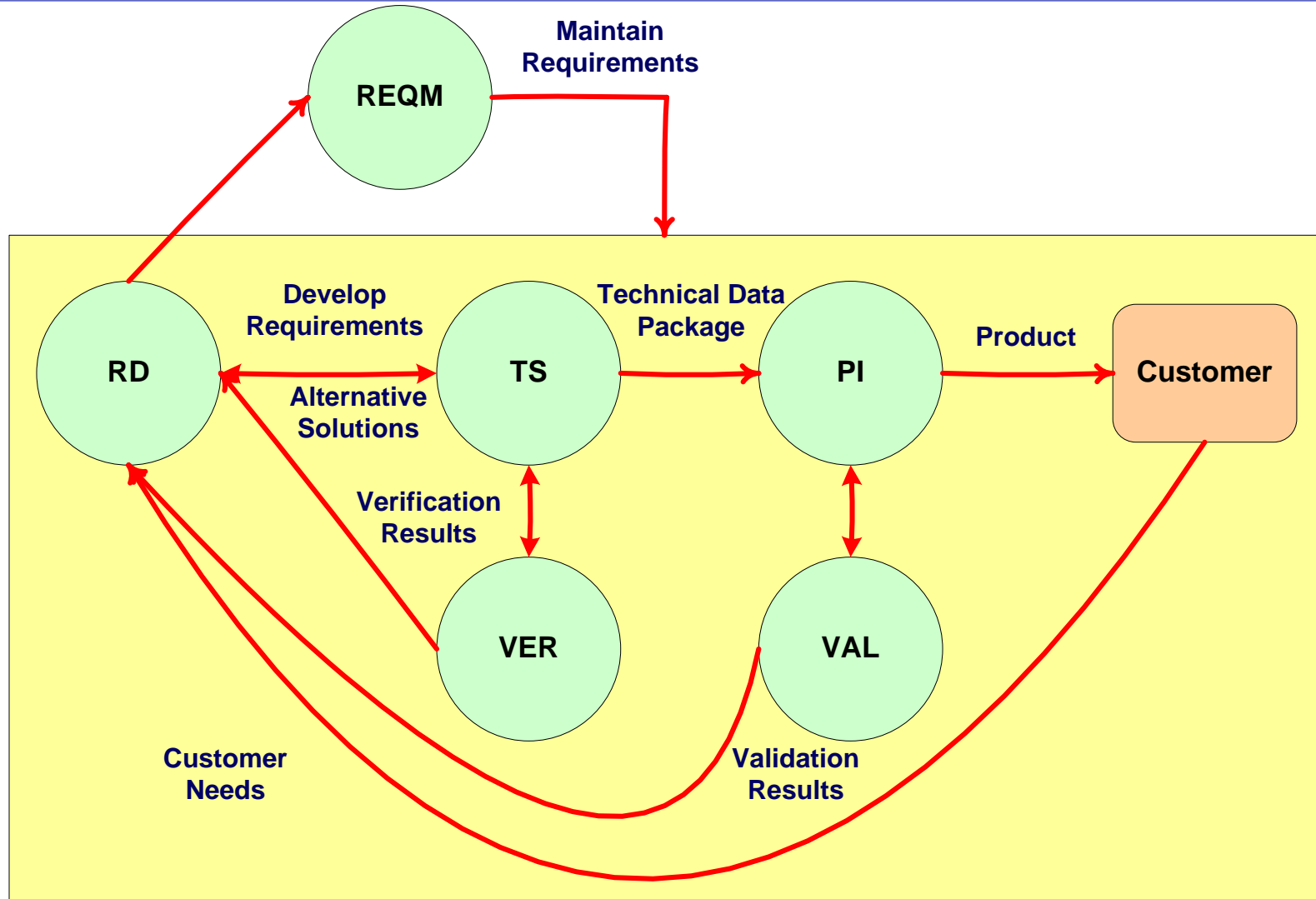


Presentation Agenda

- CMMI Engineering PA's dependencies
- Developing a Software Architecture using Attribute Driven Design (ADD)
- Technical Solution Specific Practices and how ADD activities fit in

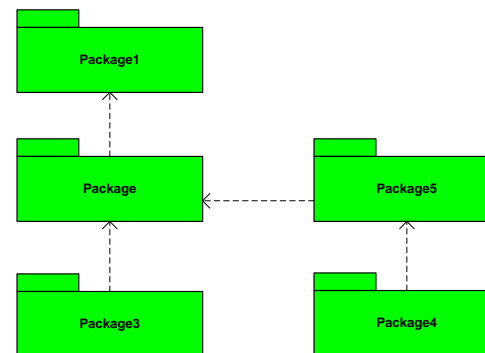


CMMI Engineering Process Areas Dependencies

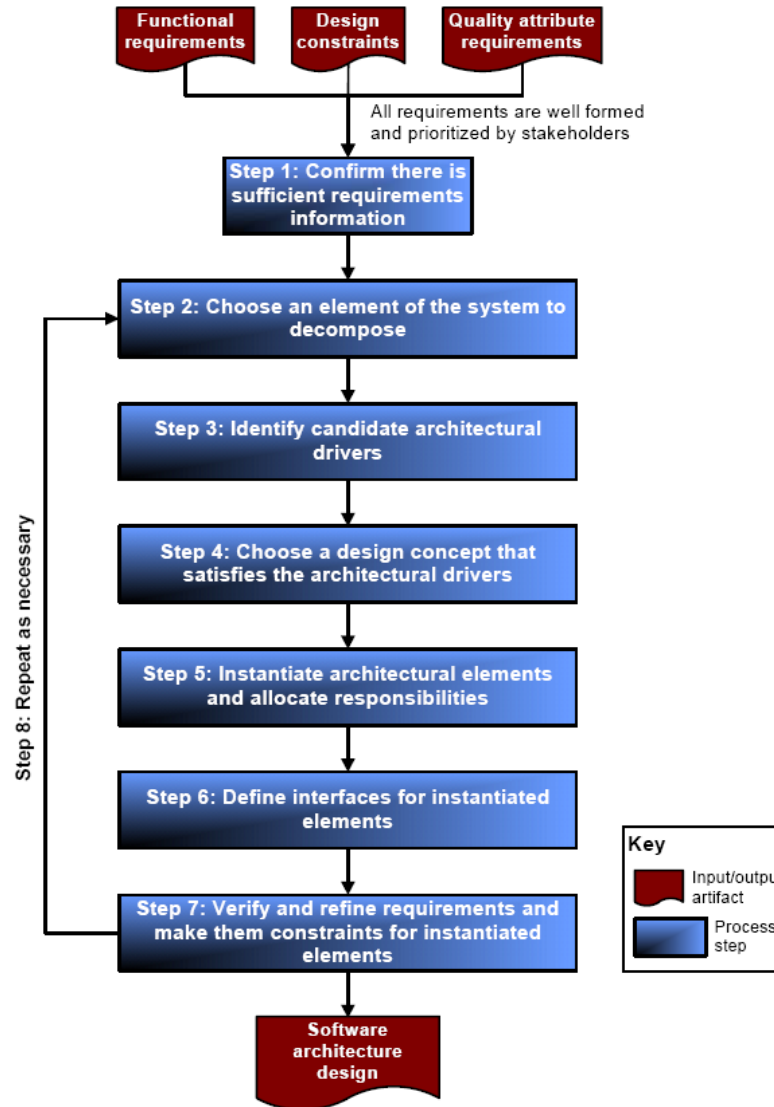


Attribute Driven Design (ADD)

- The ADD method is an approach to defining a software architecture in which the design process is based on the quality attribute requirements the software must fulfill (i.e., availability, performance, usability, security,...)
- ADD follows a recursive process that decomposes a system or system element by applying architectural tactics and patterns that satisfy its driving quality attribute requirements.
- Architectural design using the ADD methodology begins when the architectural drivers (quality attribute scenarios) are known with some level of confidence.
- Tactics and Architectural patterns are then selected to satisfy each quality attribute scenario

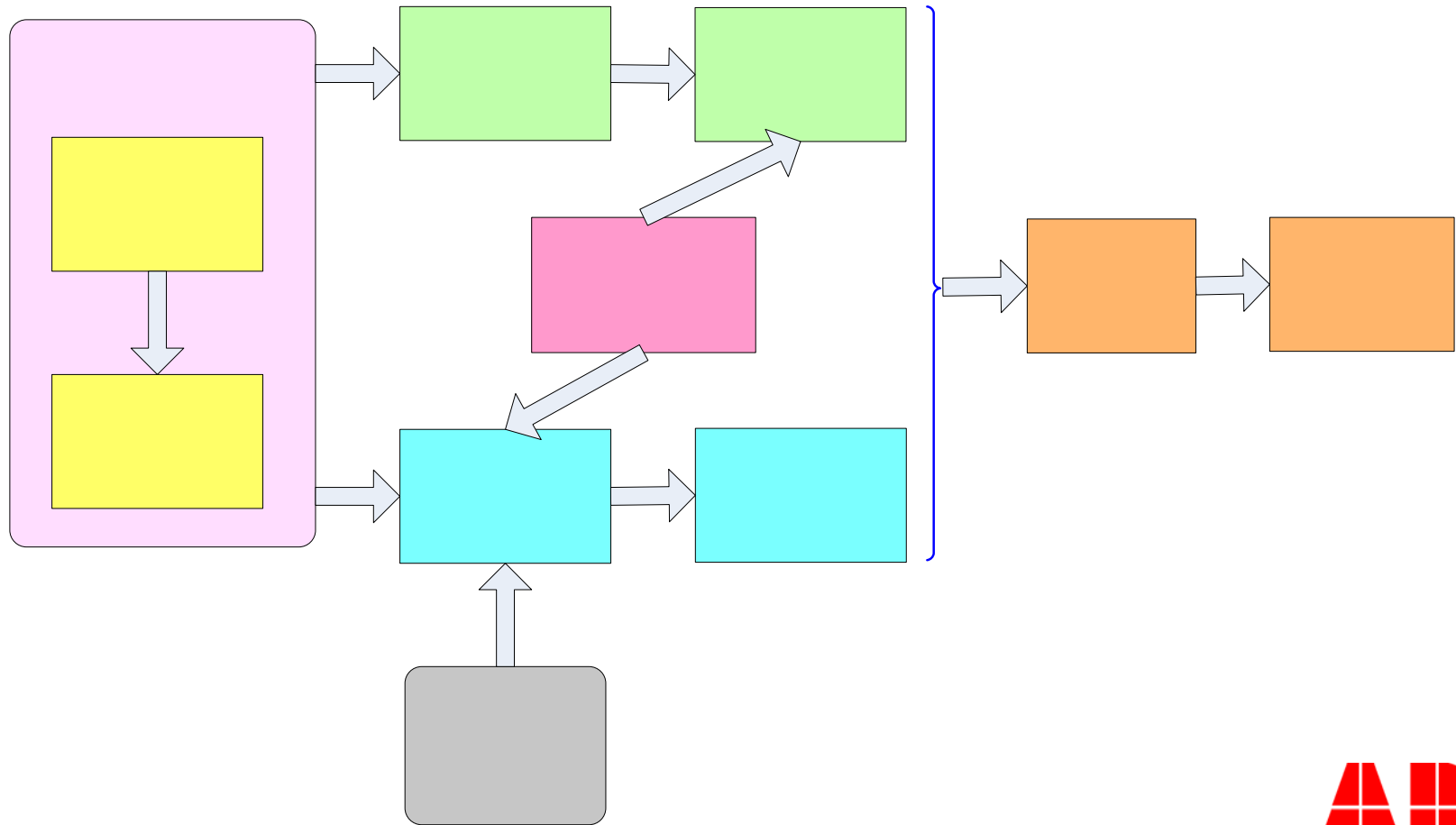


Attribute Driven Design (ADD)



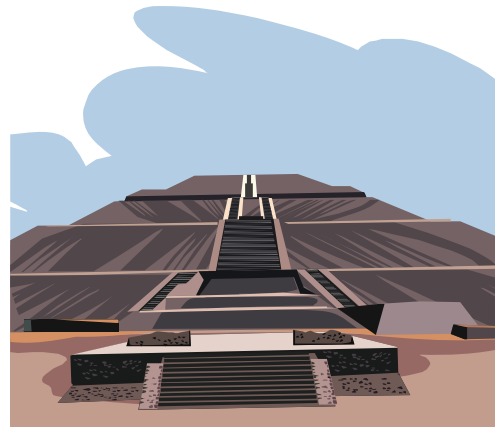
* Taken from Technical Report CMU/SEI-2007-TR-005

Integration of ADD and CMMI (TS-RD-REQM)

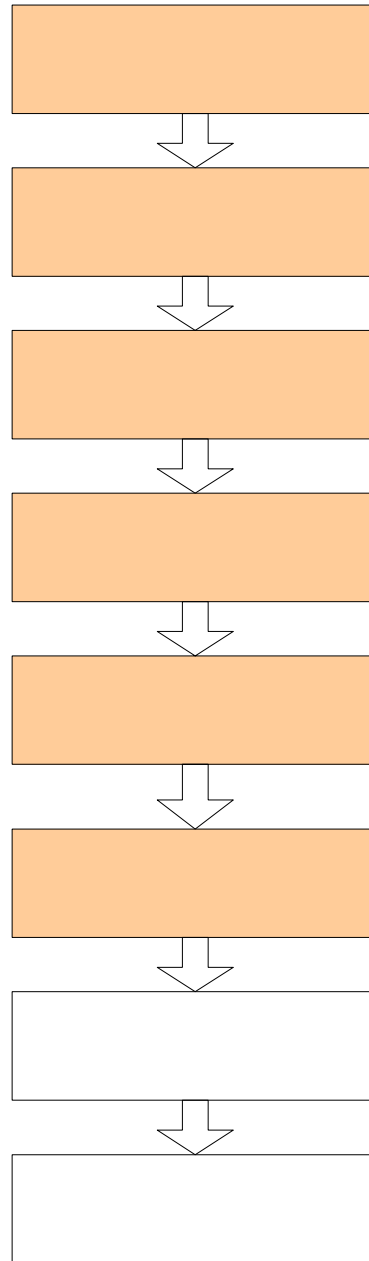


Steps for Creating a Software Architecture

- **Creating the business case for the system**
- **Understanding and documenting the requirements**
- **Leveraging Quality Attribute Scenarios**
- **Creating or selecting the architecture**
- **Documenting and communicating the architecture**
- **Analyzing or evaluating the architecture**
- **Implementing the system based on the architecture**
- **Ensuring that the implementation conforms to architecture**



Technical Solution Process Area



Specific Practice 1.1

Develop alternative solutions
and selection criteria

- The Attribute Driven Design methodology requires identification and prioritization of functional requirements, design constraints, and quality attribute requirements
- This set of requirements is the basis for the generation of architectural drivers (quality attribute scenarios)
- These quality attribute scenarios are the criteria to be used to evaluate architectural decisions

Specific Practice 1.2

Select product component solutions that best satisfy the criteria established

- Quality attribute scenarios are employed to evaluate and choose the best architectural alternatives/technologies



Specific Practice 2.1

Develop a design for the product or product component

- Document the architectural views of the product

Specific Practice 2.2

Establish and maintain a Technical Data Package

- Determine appropriate levels of design and documentation
- Document design in technical data package
- Document rationale for key design decisions

- Complete documentation of architectural views are integral part of the technical data package
- Architectural requirements and scenarios should also be included



Specific Practice 2.3

Design product interfaces
using established criteria

- Architectural diagrams should identify internal and external interfaces

Specific Practice 2.4

Perform develop, buy, or reuse analyses

- Analysis for reuse, make, or buy decisions are also important part of the decision making process

Specific Practice 3.1

Implement the product
design



Specific Practice 3.2

Develop and maintain the
end-user documentation





Power and productivity
for a better world™

