Carnegie Mellon
Software Engineering Institute

Future Directions of the Software Architecture Technology Initiative

Mark Klein

Second Annual
SATURN Workshop

April 2006

Carnegie Mellon
Software Engineering Institute

Presentation Outline

Getting (Re)acquainted

State of the SAT Initiative

Future Directions

Product Line Systems Program

The Product Line Systems (PLS) Program is one of five technical programs at the SEI.

PLS's mission: *Enable widespread product line practice and architecture-centric development throughout the global community.*

PLS's initiatives:

- **Software Architecture Technology (SAT) Initiative**
- Product Line Practice Initiative
- Predictable Assembly from Certifiable Components Initiative

Carnegie Mellon
Software Engineering Institute

What Is a Software Architecture?

"*The **software architecture** of a program or computing system is the **structure or structures** of the system, which comprise the **software elements**, the **externally visible properties** of those elements, and the **relationships among** them.*"[1]

[1] Bass, L.; Clements; P. & Kazman, R. *Software Architecture in Practice, Second Edition.* Boston, MA: Addison-Wesley, 2003.

Why is Software Architecture Important? -1

Represents *earliest* design decisions

- hardest to change
- most critical to get right
- communication vehicle among stakeholders

First design artifact addressing

- performance
- modifiability
- reliability
- security

Key to systematic *reuse*

- transferable, reusable abstraction

The right architecture paves the way for system success.

The wrong architecture usually spells some form of disaster.

Carnegie Mellon
Software Engineering Institute

Why Focus on Software Architecture?

The quality and longevity of a software system is largely determined by its architecture.

Too many experiences point to inadequate software architecture education and practices and the lack of any real software architecture evaluation early in the life cycle.

Without an explicit course of action focused on software architecture, these experiences are being and will be repeated.

The cost of inaction is too great.

Carnegie Mellon
Software Engineering Institute

Without Software Architecture Focus

Poorly designed software architectures result in

- greatly inflated integration and test costs
- inability to sustain systems in a timely and affordable way
- lack of system robustness
- in the worst case, program/system cancellation
- in all cases, failure to best support the business and mission goals

Carnegie Mellon
Software Engineering Institute

Presentation Outline

Getting (Re)acquainted

State of the SAT Initiative

Future Directions

Software Architecture Technology (SAT) Initiative's Focus

Ensure that business and mission goals are predictably achieved by using effective software architecture practices throughout the development lifecycle.

Axioms Guiding Our Work

- Software architecture is the bridge between business and mission goals and a software-intensive system.
- **Quality attribute requirements drive software architecture design.**
- Software architecture drives software development throughout the life cycle.

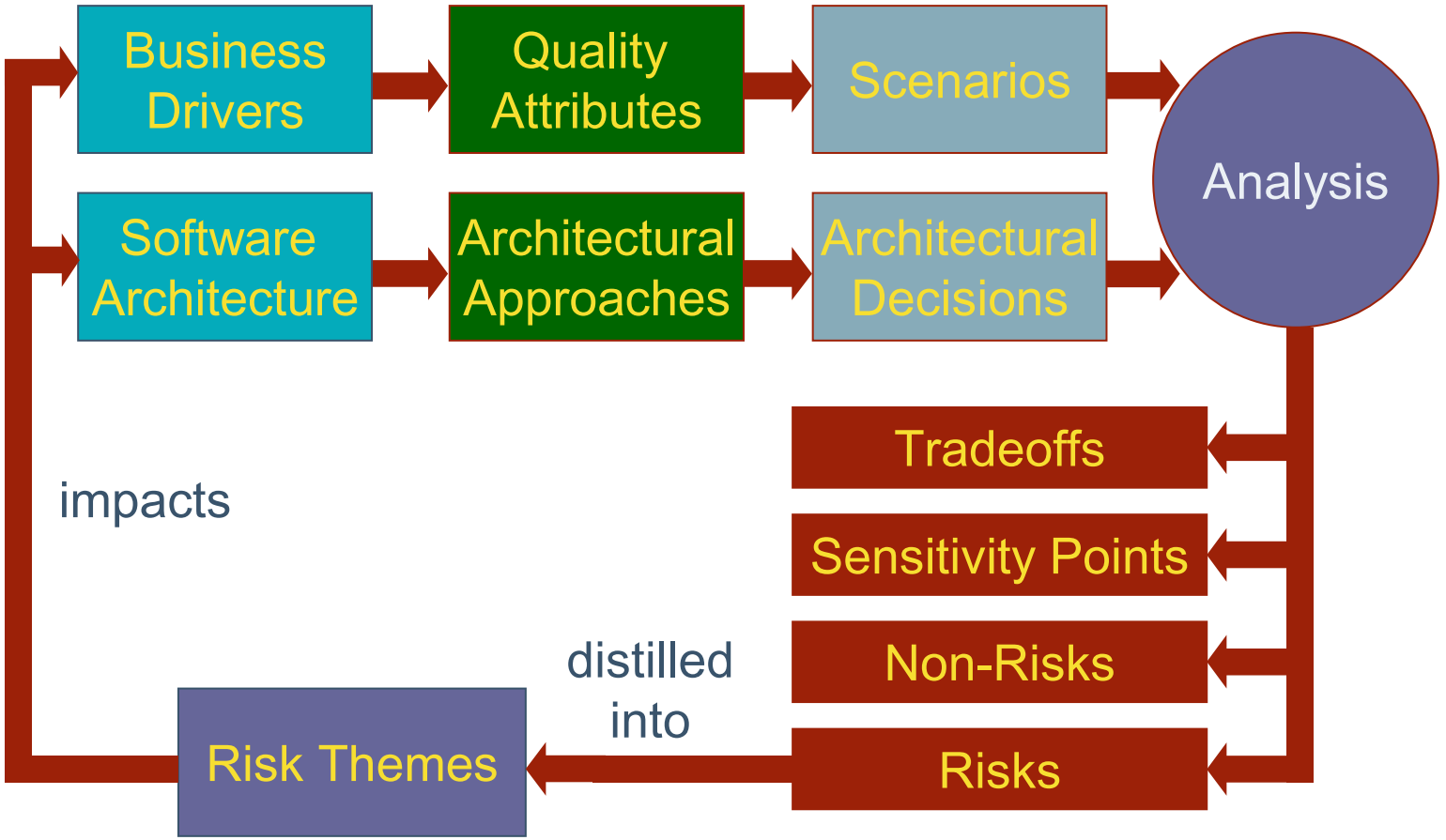
SEI's Architecture Tradeoff Analysis Method® (ATAM®)

ATAM is an architecture evaluation method that

- focuses on multiple quality attributes
- illuminates points in the architecture where quality attribute *tradeoffs* occur
- generates a context for ongoing quantitative analysis
- utilizes an architecture's vested stakeholders as authorities on the quality attribute goals

Carnegie Mellon
Software Engineering Institute

Conceptual Flow of the ATAM℠



ATAM Led to the Development of Other Methods and Techniques

What if the quality requirements are not well-understood?

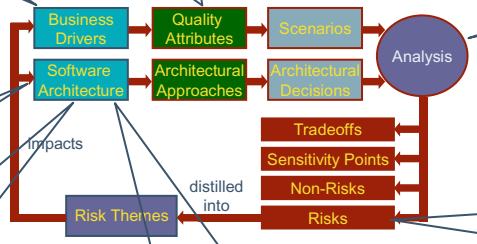
Quality Attribute Workshop

Our scenarios tend to be incomplete or ambiguous.

Quality Attribute General Scenarios

What are some of the most important questions to ask?

Quality Attribute Tactics



What if there's no architecture?

Attribute Driven Design

Which risks should I work on first?

Cost Benefit Analysis Method

What if I don't know my system's architecture?

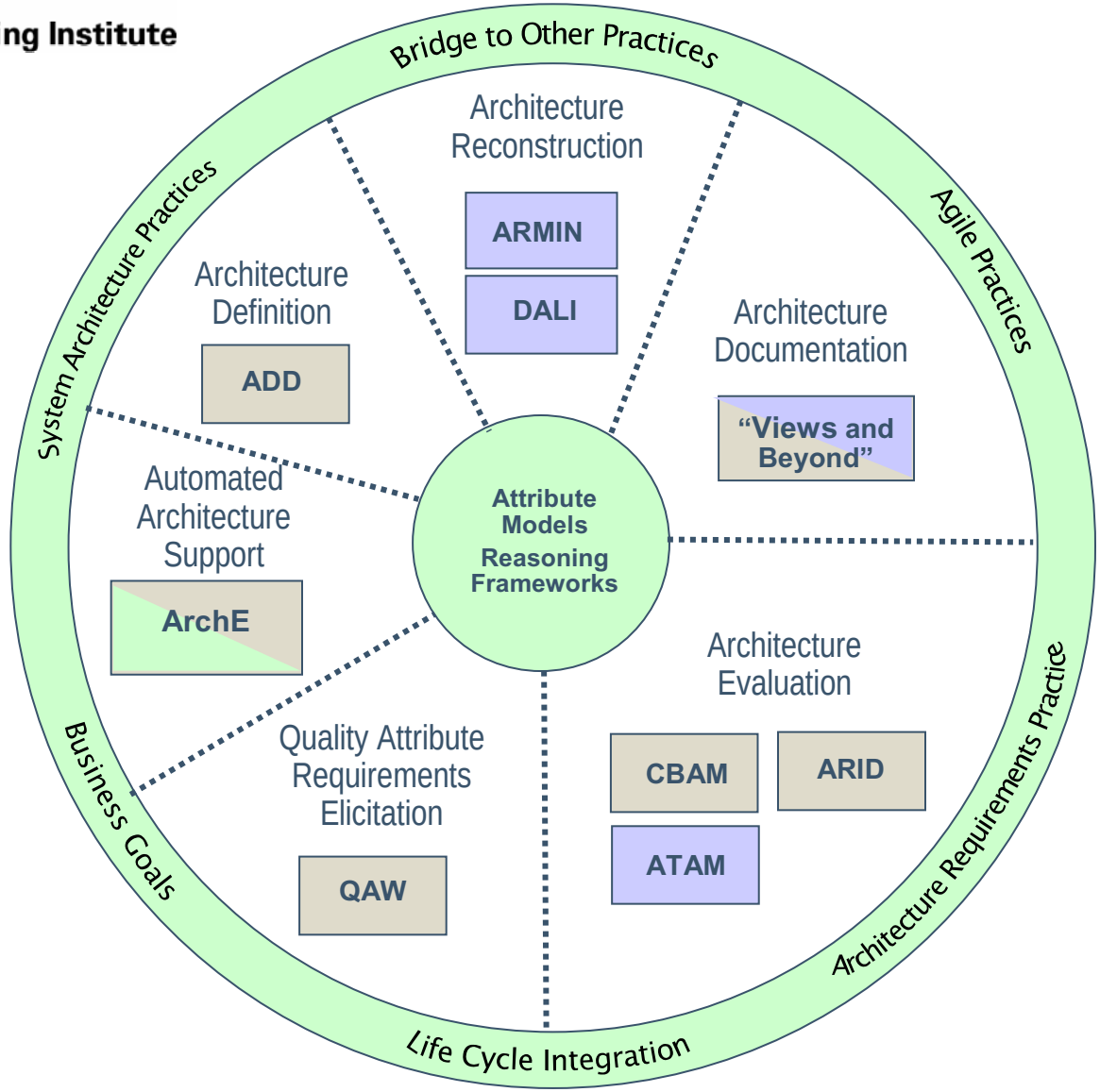
Architecture Reconstruction using ARMIN

What information should be included in my architecture documentation?

Views and Beyond Approach

Carnegie Mellon
Software Engineering Institute

SAT: Technology



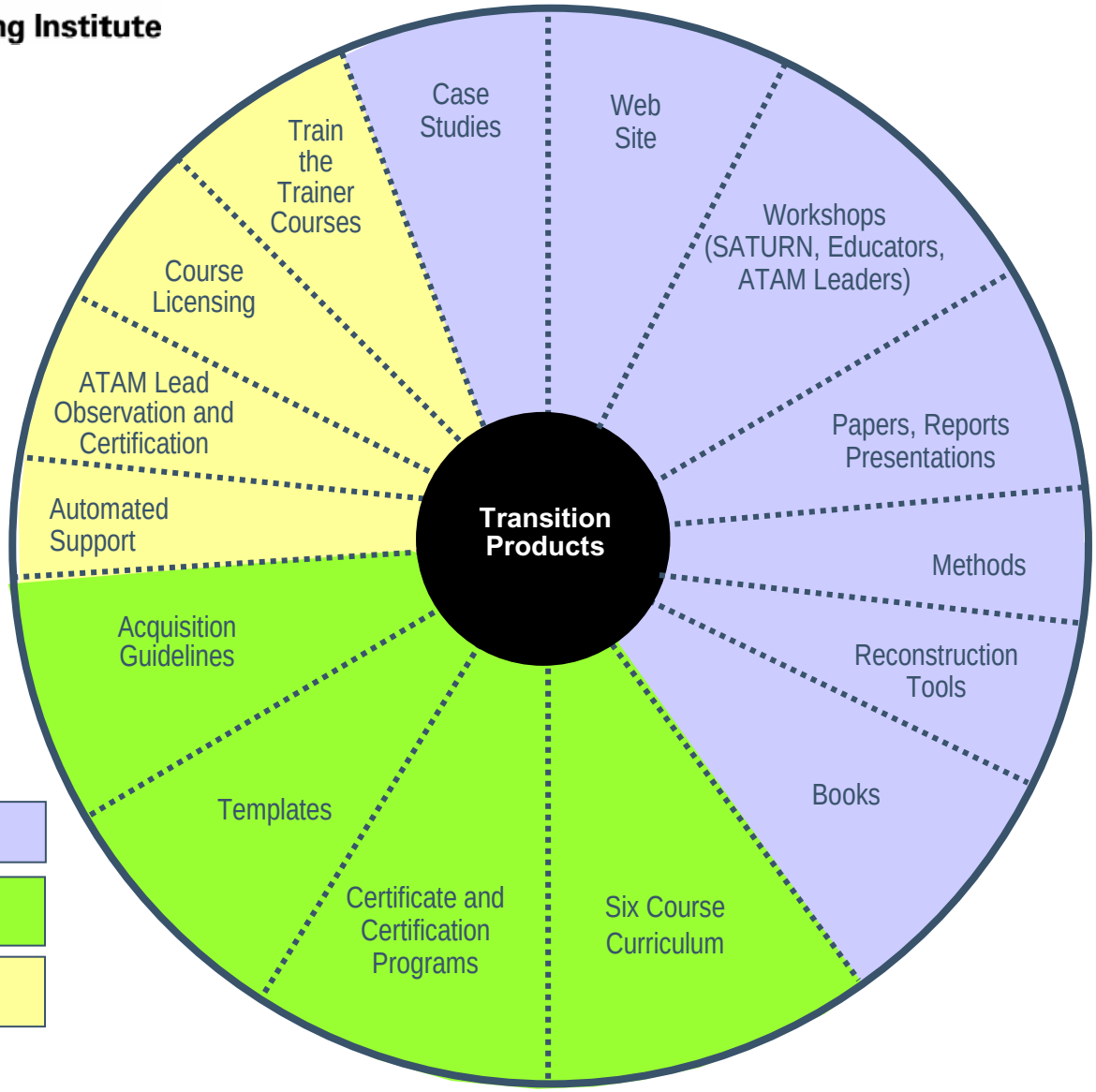
In Transition

In Application

In Research

Carnegie Mellon
Software Engineering Institute

SAT: Transition



Ongoing

In Sustainment

Just Begun

Carnegie Mellon
Software Engineering Institute

Other Recent Work

Showed how to use aspects for architecture enforcement

Investigated categorization of business goals accumulated from ATAM evaluations

Building tradeoff analysis into ArchE

Conducted first annual ATAM Lead Evaluator Workshop

Launched licensing of the Software Architecture Principles and Practices Course

Carnegie Mellon
Software Engineering Institute

Work "Hot Off the Press"

"A Comparison of Requirements Specification Methods from a Software Architecture Perspective"

- What does it mean for a requirements document to be really what an architect needs?
- What do the existing requirement specification methods offer in capturing architecturally significant requirements?

Business goal and risk theme analysis

- Based on examining data collected from ATAM evaluations, is there a useful categorization of business goals and risk themes?
- Is there a correlation between business goal categories and risk theme categories?
- What is a useful data collection and analysis methodology for analyzing the results of the ATAMs?

Carnegie Mellon
Software Engineering Institute

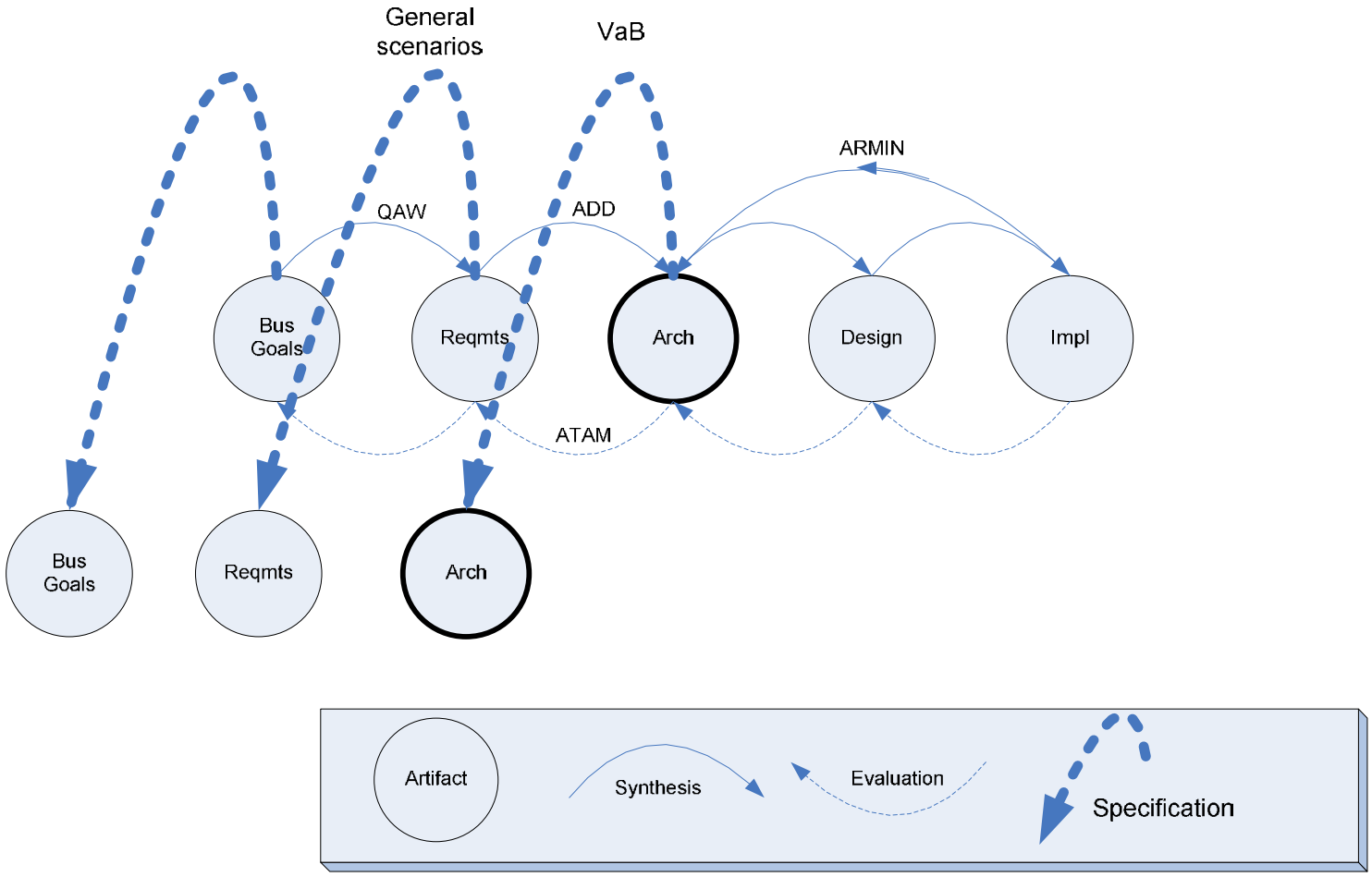
Presentation Outline

Getting (Re)acquainted

State of the SAT Initiative

Future Directions

Architecture-Centric Life-Cycle Practices -1



Architecture-Centric Life-Cycle Practices -2

In support of the SAT axioms:

- Since *"Software architecture is the bridge between mission/business goals and a software-intensive system",* we need to better understand
  - The relationship between business goals and quality attribute requirements
  - How to specify business goals

- Since *Software architecture drives software development through the life cycle",* we need to better understand
  - Refining architectures to detailed designs
  - Techniques for ensuring that detailed designs conform to the architectures

Architecture Evolution -1

Since "*Quality attribute requirements drive software architecture design*" and "*Software architecture is the bridge between mission/business goals and a software-intensive system*":

- The quality and longevity of a software system is largely determined by its architecture.
- Therefore a system's software architecture offers leverage for ensuring that a system continues serving an organization's business as those goals evolve.

Architecture Evolution -2

Evolution requires making architectural decisions under uncertainty:

- Responding to change effectively while maximizing value-added using notions from utility theory
- Exploiting theories such as *real options* theory to place a value on flexibility
- Exploiting quality attribute theories to make sound quality decisions
- “Optimizing” the timing of and trade-offs in design decisions

Architecture Evolution -3

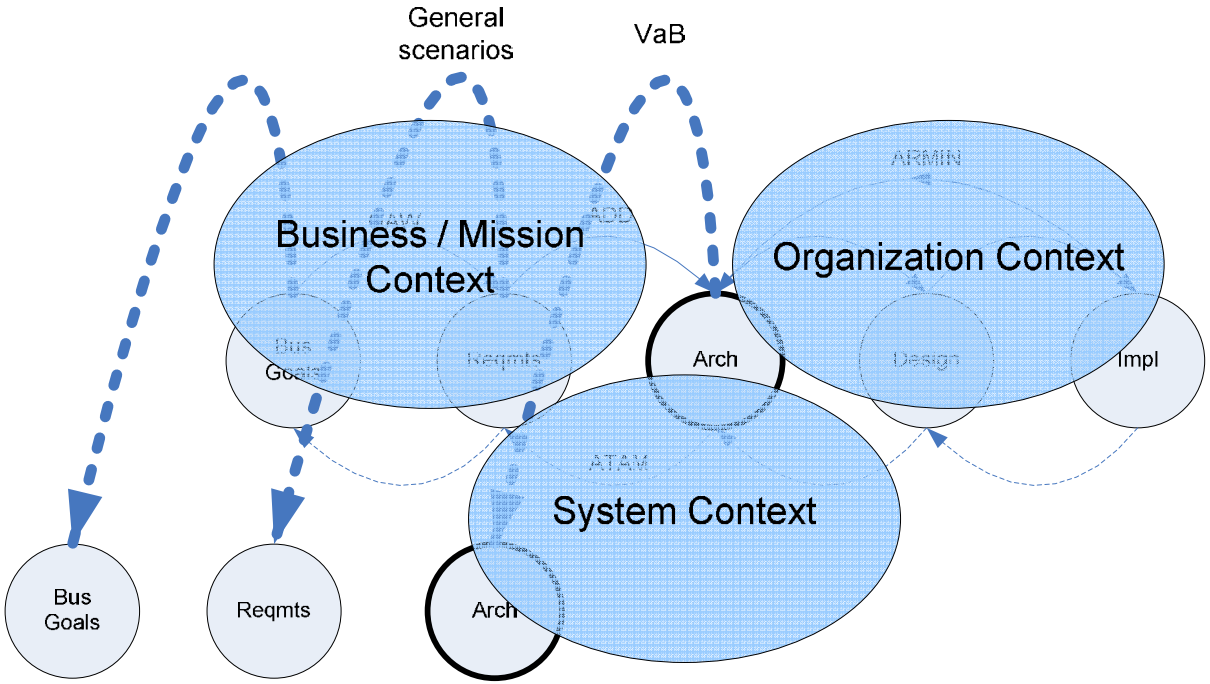
Methodological support

- Defining *practical* and *economics-driven* methods for valuing architectural decisions in relation to quality attributes. The Architecture Improvement Workshop (AIW) is a starting point.
- Tying together existing methods such as QAW, ADD, ATAM and CBAM

Augmenting methods with automation support

- ArchE – automated architectural design assistant
- ARMIN – architecture reconstruction
- Prototype documentation environment

Architecture Practice in Context



Carnegie Mellon
Software Engineering Institute

Architecture Competence

What does an organization need to do to model, measure and improve its competence in performing architecture-centric software engineering?

- What are the skills that enable a competent architect such as technical, social, leadership skills and situation-specific skill profiles?
  - *We plan to start by conducting structured interviews and surveys*

- How do you systematically capture organizational experience?
  - *We plan to build simple models using checklists and then look at Design for Six Sigma*

- What's the relationship between organizational structures and architectural structures?
  - *We plan to build on the results of an SEI IR&D investigating communication patterns vis-à-vis architectural dependencies*

Carnegie Mellon
Software Engineering Institute

Systems Architecture Practices

How can we close the gap between the engineering practices of system architecture and software architecture?

- How do you manage the system's quality attributes within and between the system and software architecture(s)?

- How do you describe the mapping between the operational architecture, system architecture and software architecture representations? How do you relate the views in the architectures?

Game Plan

- Use structured interviews to assess state of the practice
- Augment current methods to account for system architecture practices

Carnegie Mellon
Software Engineering Institute

Architecture Technology

We plan to continue investigating technologies such as

- Service Oriented Architectures
- Aspect-oriented design

As systems continue to get larger and more complex does the nature of architecture change?

- We intend to investigate potentially applicable techniques from areas such as
  - Mechanism and institutional design
  - Self-adaptive systems
  - Complex adaptive systems

Carnegie Mellon
Software Engineering Institute

SAT "Axioms" and New Directions

"Axioms" Guiding Our Work

- Software architecture is the bridge between mission/business goals and a software system
- Software architecture drives software development throughout the life-cycle.
- Quality attribute requirements drive software architecture design.

New Directions

- Expand current work from design and development to also address system evolution
- Investigate architectural competence
- Investigate the use of economic models, various theories of design, and theories from other disciplines
- Investigate the nature of architecture as systems become ultra-large

Carnegie Mellon
Software Engineering Institute

We want your input!

Our ongoing goals are to

- Respond to the needs of the world
- Increase our level of impact
- Base techniques and methods on theoretically sound foundations

We are very much looking forward to getting your thoughts!