Analyzing Quality Attributes as a Means to Improve Acquisition Strategies

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Lisa Brownsword, Patrick Place,
Cecilia Albert, David Carney

May 2014

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN “AS-IS” BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

DM-0001274

Purpose of Our Research

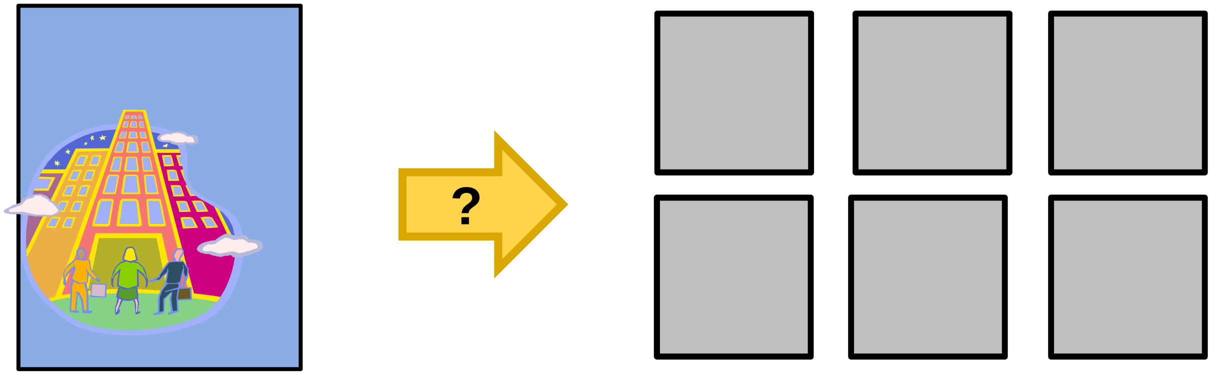
Investigate the utility of a method that produces mutually constrained and aligned program acquisition strategy and system and software architecture to improve the probability of a program’s success; the method is to be used by PMOs for software-sensitive programs

Why this is important

- Software is increasingly important to the success of government programs
- There continues to be little consideration of the software architecture in the development of either the system architecture or the program’s acquisition strategy
- Software architecture is often over constrained by decisions made early in the acquisition lifecycle when key program choices are being made – negatively affecting program success.

Alignment among the software and system architecture and acquisition strategy does not occur naturally

Interplay of Acquisition and Architecture



monolithic legacy architecture

new modular architecture with new and legacy capabilities

Program Manager

Should I have 1 contractor, or 2 or 3 or 6?

If 1 contractor, how do I enforce a modular architecture?

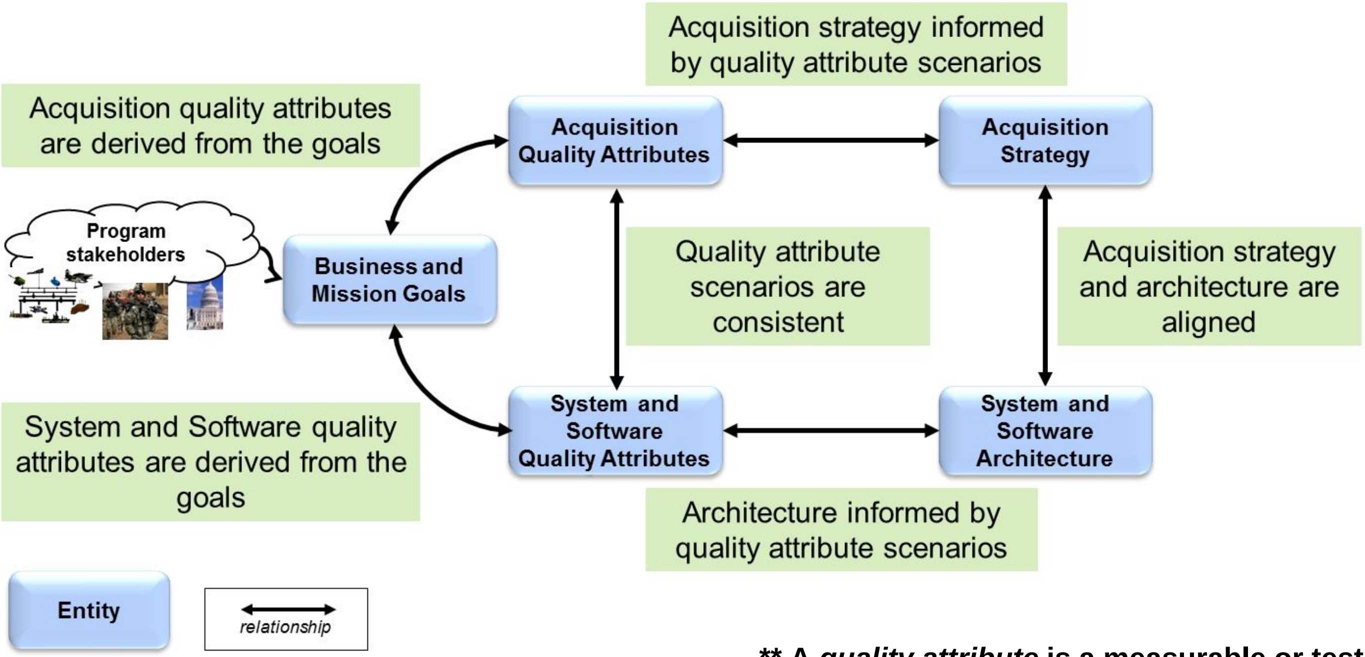
If multiple contractors, how do I ensure the parts fit together?

Can I migrate legacy to give me a quick delivery?

Our Hypothesis

If sufficient business and mission goals are elicited from program stakeholders, they can be used to create an Acquisition Strategy, System Architecture, and Software Architecture that are mutually aligned – thus avoiding a common pattern of program failure*.

The principal mechanisms used to accomplish this are *quality attributes***.



Acquisition strategy informed by quality attribute scenarios

Acquisition quality attributes are derived from the goals

Acquisition Quality Attributes

Acquisition Strategy

Program stakeholders

Business and Mission Goals

Quality attribute scenarios are consistent

Acquisition strategy and architecture are aligned

System and Software quality attributes are derived from the goals

System and Software Quality Attributes

System and Software Architecture

Architecture informed by quality attribute scenarios

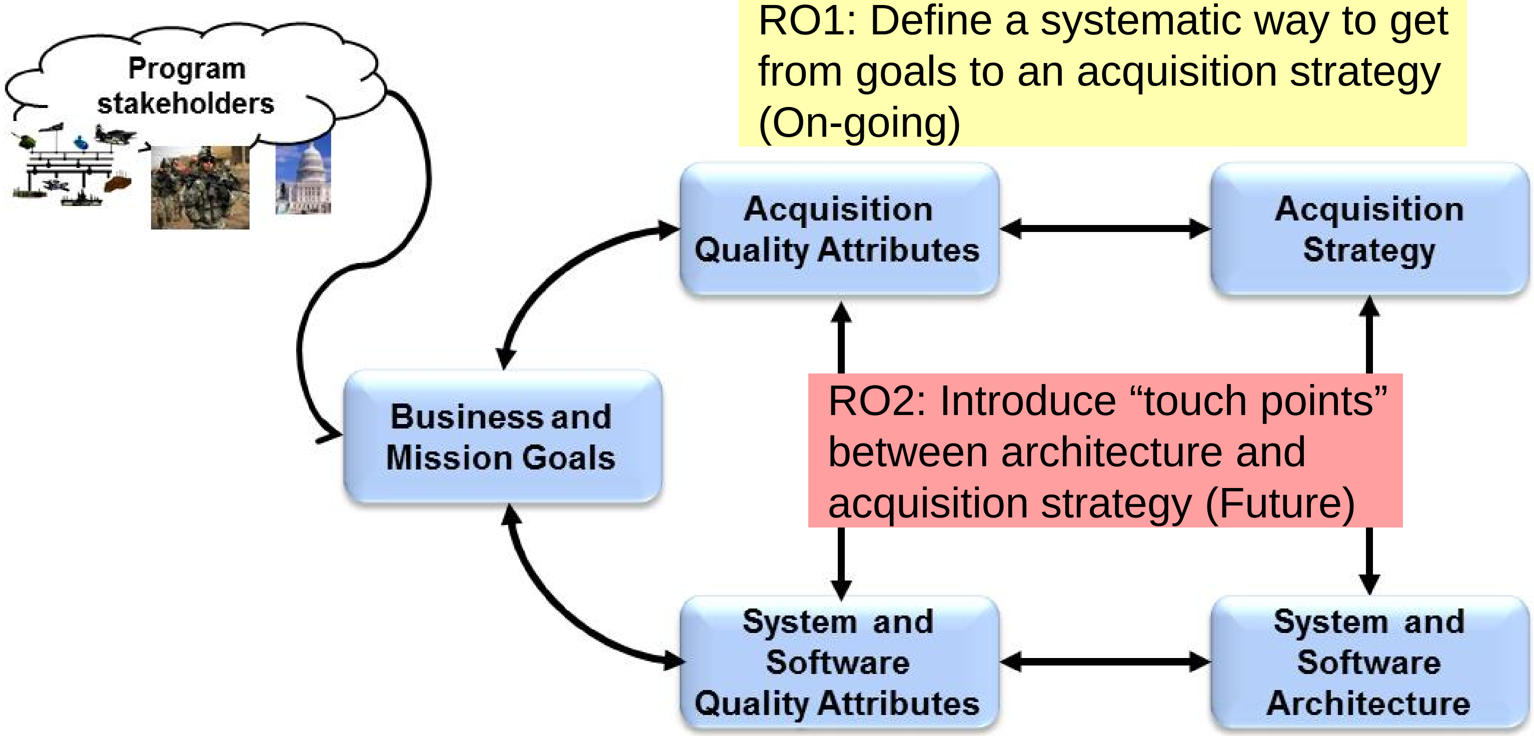
Entity

relationship

** A ***quality attribute*** **is a measurable or testable property that is used to indicate how well the system, software, or program satisfies the needs of its stakeholders.**

* Phase 1 results published in SEI TN CMU/SEI-2013-TN-014: “Isolating Patterns of Failure in Department of Defense Acquisition“

Research Opportunities from Phase 1

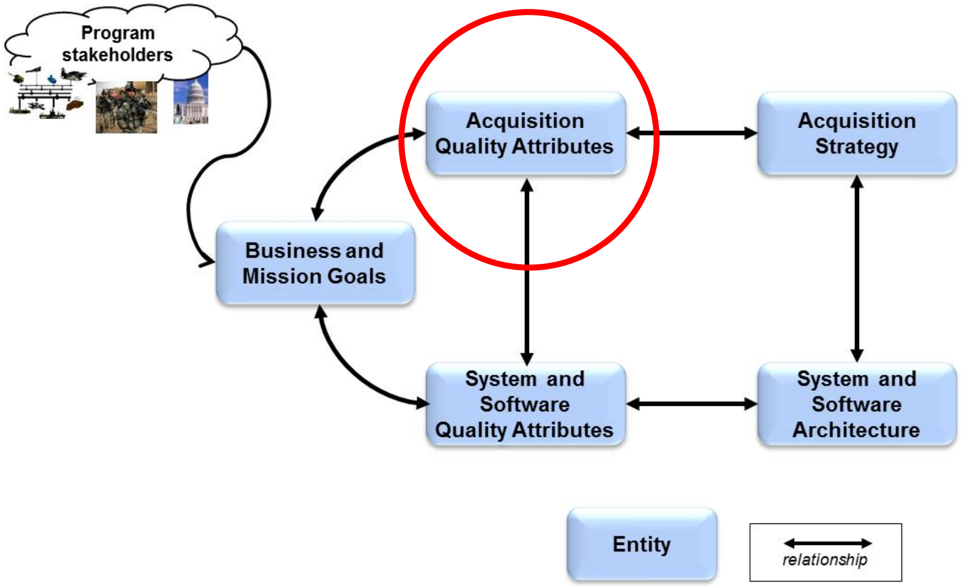

Program stakeholders
Acquisition Quality Attributes
Acquisition Strategy
Business and Mission Goals
System and Software Quality Attributes
System and Software Architecture

RO1: Define a systematic way to get from goals to an acquisition strategy (On-going)

RO2: Introduce “touch points” between architecture and acquisition strategy (Future)

Nearly 20 years of experience reflecting goals, through quality attributes, to system and software architectures (Complete)

Phase 2: Explore *Acquisition* Quality Attributes


Program stakeholders
Acquisition Quality Attributes
Acquisition Strategy
Business and Mission Goals
System and Software Quality Attributes
System and Software Architecture
Entity
relationship

Captured 75 scenarios across 23 programs

- Identify candidate acquisition quality attributes (AQA)
- Determine how to express program-specific AQAs
- Construct and analyze AQA scenarios
- Build a prototype workshop to elicit AQA scenarios

Phase 2 results published in SEI TN CMU/SEI-2013-TN-026:
“Results in Relating Quality Attributes to Acquisition Strategies“

Candidate Acquisition Quality Attributes

Original candidates

Acceptability
Accountability
Affordability
Appropriateness of contract
Appropriateness of technology
Achievability
Accreditability
Balance
Commitability
Communicability
Competitiveness
Contract manageability
Credibility
Effectiveness
Evolvability
Fairness
Flexibility
Implementability
Legality
Manageability of risk
Management visibility
Modifiability
Promptness in reporting problems
Responsibility
Responsiveness
Sensibility
Staffability
Suitability
Sustainability
Timeliness
Traceability with requirements

Sources: DoD acquisition strategy guidance and instruction documents

What our data showed

Acquisition Quality Attribute	Frequency
Flexibility	23
Performability	15
Realism	14
Affordability	10
Survivability	6
Executability	5
Responsiveness	4
Programmatic Transparency	2
Innovativeness	1
Schedulability	1

Phase 2 Findings

Expressing AQA scenarios similarly to software QA scenarios is a viable path

Software QA Scenarios	Acquisition QA Scenarios
Software architecture	Acquisition strategy
System	Program
Architect	Program manager

Scenario from software domain:

Stimulus:	An internal component fails
Environment:	During normal operation
Response:	The system is able to recognize a failure of an internal component and has strategies to compensate for the fault

Scenario from acquisition domain:

Stimulus:	An unexpected budget cut
Environment:	For a multi-segment system
Response:	The program is able to move work between major segments to speed up or slow down separate segments within the available funding

Value of AQA scenarios

AQA scenarios can be used to

- Express effects of business and mission goals
- Inform the development of the acquisition strategy
- Determine appropriateness of acquisition strategy with respect to any given scenario

Acquisition Quality Attribute	Scenario	Potential Acquisition Tactic
Flexibility	The user’s system requirements change radically 30 days before the RFP is released when the “go live” date is fixed; the RFP is released regardless.	Establish fallback strategies that protect the “go live” date.
Affordability	We discover that the cost of operating the system will be higher than the ceiling mandates during development but before initial fielding; the system (including its architecture) is shifted to a less costly alternative.	Emphasize the need for architecture adaptability and flexibility.

Incompatibilities between Scenarios

Stakeholder A: advocates use of open architecture as a means of avoiding vendor lock; reduce life cycle costs; increase response to user needs

Stimulus	Users request significant new functionality to be delivered rapidly
Environment	During the program's development phase
Response	Create the functionality rapidly by reusing open source, commercial-off-the shelf, and software from other projects to provide much of the capability.

Stakeholder B: is responsible for ensuring that the deliverables meet rigorous safety standards

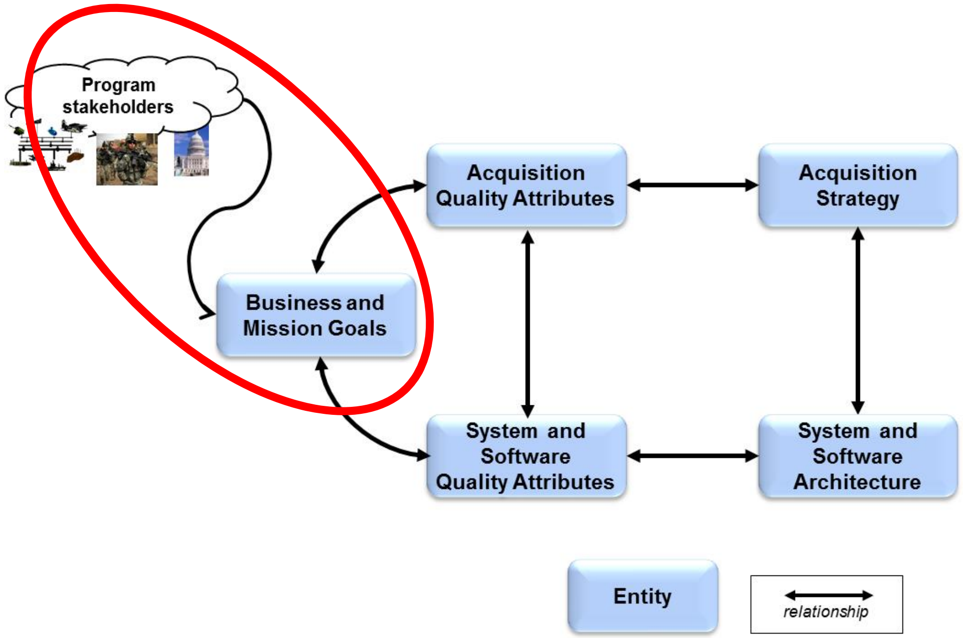
Stimulus	A new requirement to adhere to a rigorous safety standard is applied to the system
Environment	During the program's development phase
Response	The developers remove all unreachable code to insure that the system will pass stringent new certification standards.

Phase 3: Develop and Pilot an Method

Research questions that are focusing our work this year

- Can business goals that represent the full range of program stakeholders be explicitly defined and prioritized?
- Will having a more complete, explicit set of business goals generate a more complete set of AQA scenarios?
- Can having a more complete set of AQA scenarios lead to better acquisition strategies?
- Will a more systematic method for reflecting stakeholder goals in the program's acquisition strategy be useful to a program?

Phase 3a: Business Goal Determination


Program stakeholders
Acquisition Quality Attributes
Acquisition Strategy
Business and Mission Goals
System and Software Quality Attributes
System and Software Architecture
Entity
relationship

Focus on capturing business and mission goals

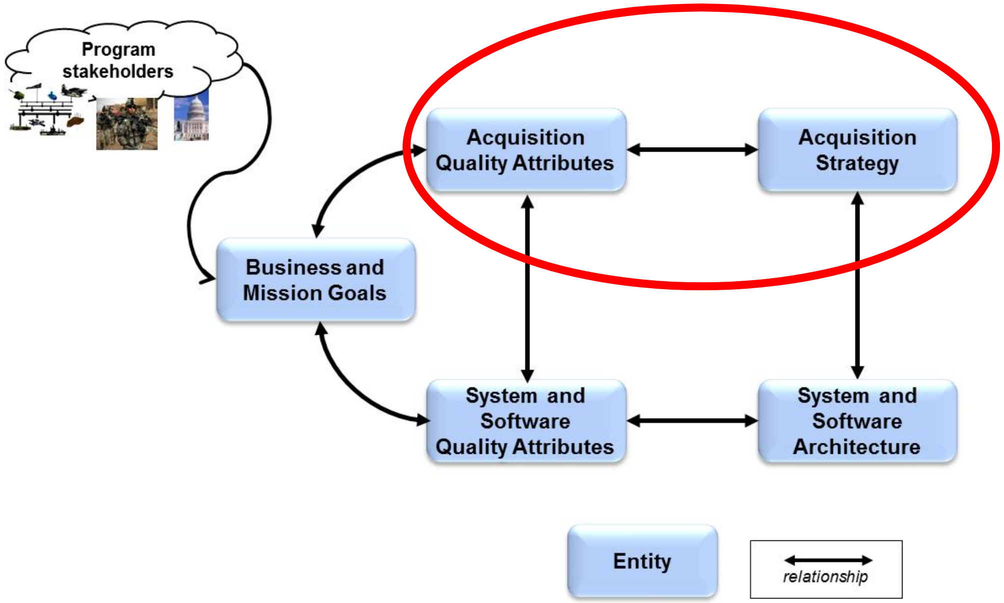
- Identify stakeholders
- Elicit business goals
- Represent goals in standard form*

Analyze goal subjects and objects to identify additional stakeholders

Note that this also applies for elicitation of mission goals

*Business Goal Scenarios found in SEI TN CMU/SEI-2010-TN-018: “Relating Business Goals to Architecturally Significant Requirements for Software Systems“

Phase 3b: Quality Attribute Consistency


Program stakeholders
Acquisition Quality Attributes
Acquisition Strategy
Business and Mission Goals
System and Software Quality Attributes
System and Software Architecture
Entity
relationship

Focus on the relationship between AQA scenarios and acquisition strategy

- Providing intuition about the AQAs
- Defining types of scenarios that might occur for a given AQA
- Creating acquisition strategy tactics associated with AQAs

Relationship to Open Architecture

- Largely, our focus is orthogonal to the question of whether or not there is an open architecture
- The value of our method is to foster explicit, program-specific, discussion of the goals that are behind the desire for an OA
  - Allows for more reasoned analysis and tradeoffs among the goals
  - Assists in ensuring that the OA business goal is supported in the acquisition strategy
- By developing AQA scenarios based on all of the goals, conflicts that require resolution can be made visible:
  - One implied business goal might be to avoid vendor lock
  - Another business goal might be to develop the system within 6 months
  - Another goal might be that the system adhere to stringent new security standards just emerging from OSD
- A likely scenario is that components from only one vendor meet the new standards; this implies a conflict between #1 avoiding vendor lock and #2 developing the system rapidly

Conclusion

We are making progress on defining the initial method steps for RO1

We are looking for candidate programs to pilot the method

There is more work beyond this year’s effort

- Extend method based on pilots
- Study the relationship between acquisition strategy and architecture
- Determine how to make scenarios consistent with each other
- Create an assessment instrument to judge alignment of acquisition strategy, system architecture, and software architecture
- Develop metrics to determine effectiveness of the method

Contact Information

Lisa Brownsword

Client Technical Solutions
Telephone: +1 703-247-1383
Email: llb@sei.cmu.edu

Patrick Place

Client Technical Solutions
Telephone: +1 412-268-7746
Email: prp@sei.cmu.edu

Cecilia Albert

Client Technical Solutions
Telephone: +1 703-247-1369
Email: cca@sei.cmu.edu

David Carney

Client Technical Solutions
Telephone: +1 505-474-2950

U.S. Mail

Software Engineering Institute
Customer Relations
4500 Fifth Avenue
Pittsburgh, PA 15213-2612 USA

Web

www.sei.cmu.edu
www.sei.cmu.edu/contact.cfm

Customer Relations

Email: info@sei.cmu.edu
SEI Phone: +1 412-268-5800
SEI Fax: +1 412-268-6257

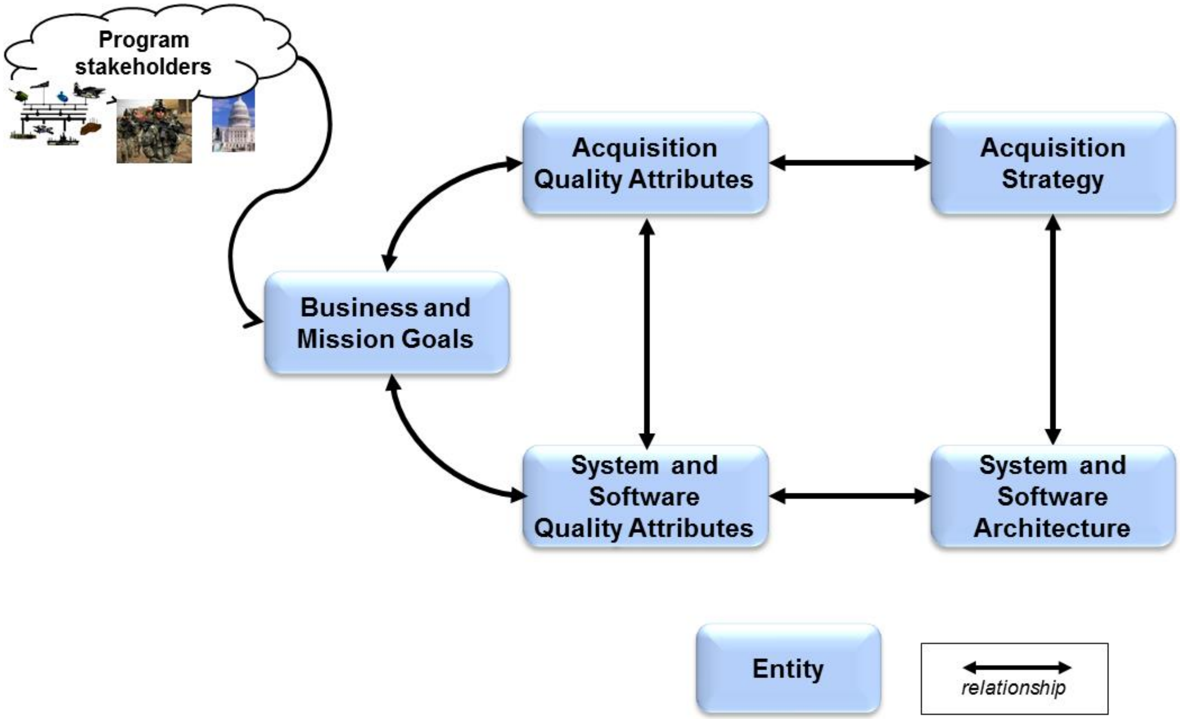
BACKUP

Phase 1: Characterize Failure Patterns

Recurring patterns of failure

- Undocumented Business Goals
- Poor Consideration of Software
- Unresolved Conflicting Goals
- Failure to Adapt
- Turbulent Acquisition Environment
- Overlooking Quality Attributes
- Inappropriate Acquisition Strategies

Entities and relations: the way it should be


Program stakeholders
Acquisition Quality Attributes
Acquisition Strategy
Business and Mission Goals
System and Software Quality Attributes
System and Software Architecture
Entity
relationship

Phase 1 results published in SEI TN CMU/SEI-2013-TN-014:
“Isolating Patterns of Failure in Department of Defense Acquisition“

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN “AS-IS” BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

DM-0001274