Architectures in Context:
A Scenario-Based Lifecycle

Hadar Ziv
ziv@ics.uci.edu

In cooperation with:
Prof. Debra Richardson
Prof. Thomas Alspaugh
Prof. Thomas Standish
And the ROSATEA group at UC Irvine

Sergio Alvarado
Stewart Sutton
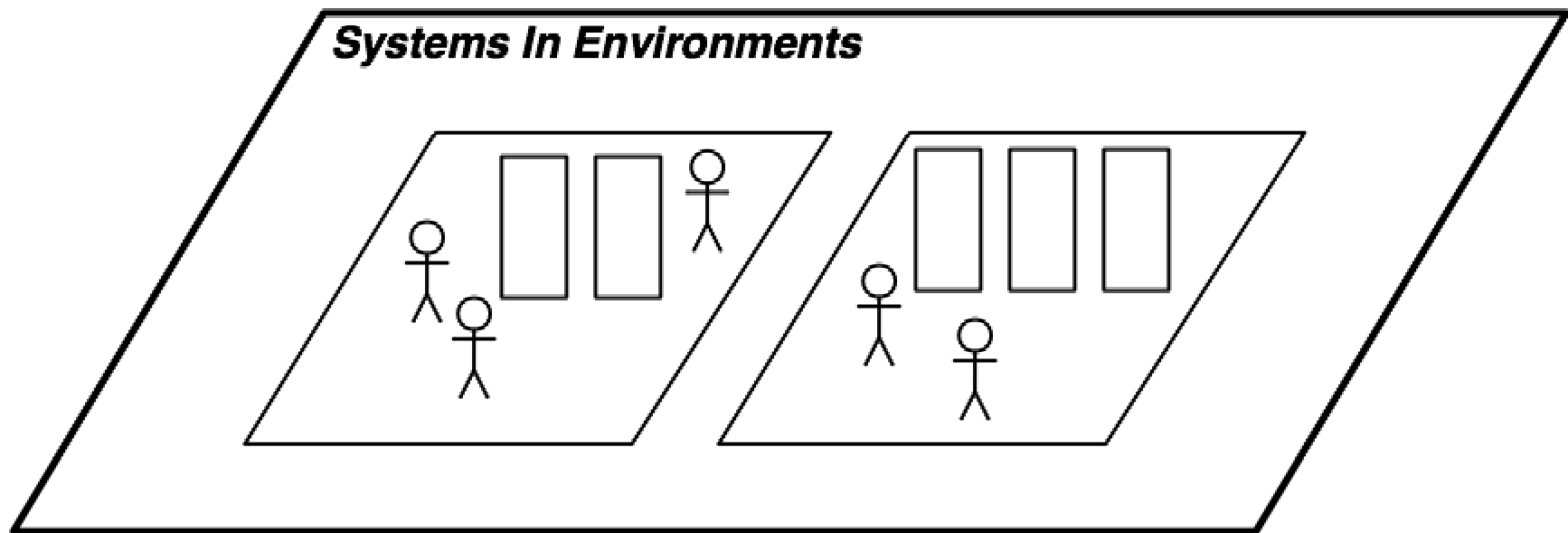And the KRISP team at Aerospace

Motivation

- Some terms/concepts are loosely defined
  - Use cases
  - Scenarios
  - Goals

- Big nasty problems persist
  - Scalability
    - Family of applications, product families
    - Enterprise architectures, product-line architectures
    - Systems of systems
  - Traceability
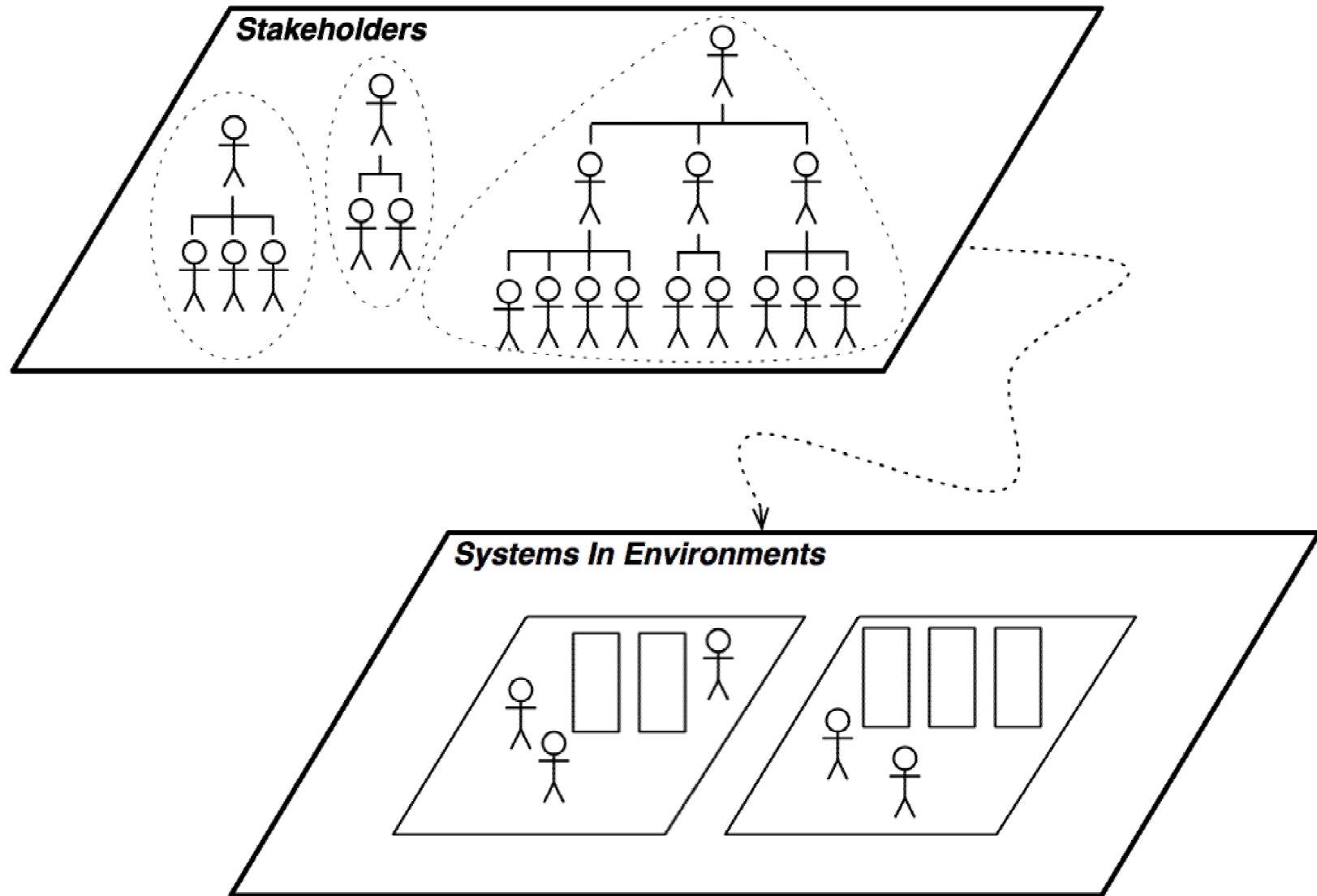 - Testing

Question Everything!

- ❖ Question the basic definitions of relationships between
  - ❖ Use cases to *Goals* (1:1?)
  - ❖ Use cases to *Scenarios* (1:m?)
  - ❖ *Goals* to *Scenarios*?
  - ❖ All of the above to design and implementation???
- ❖ Question the process model/product development lifecycle
  - ❖ We have defined a lifecycle model focused on artifacts
  - ❖ It is high-level enough to be flexible and scaleable
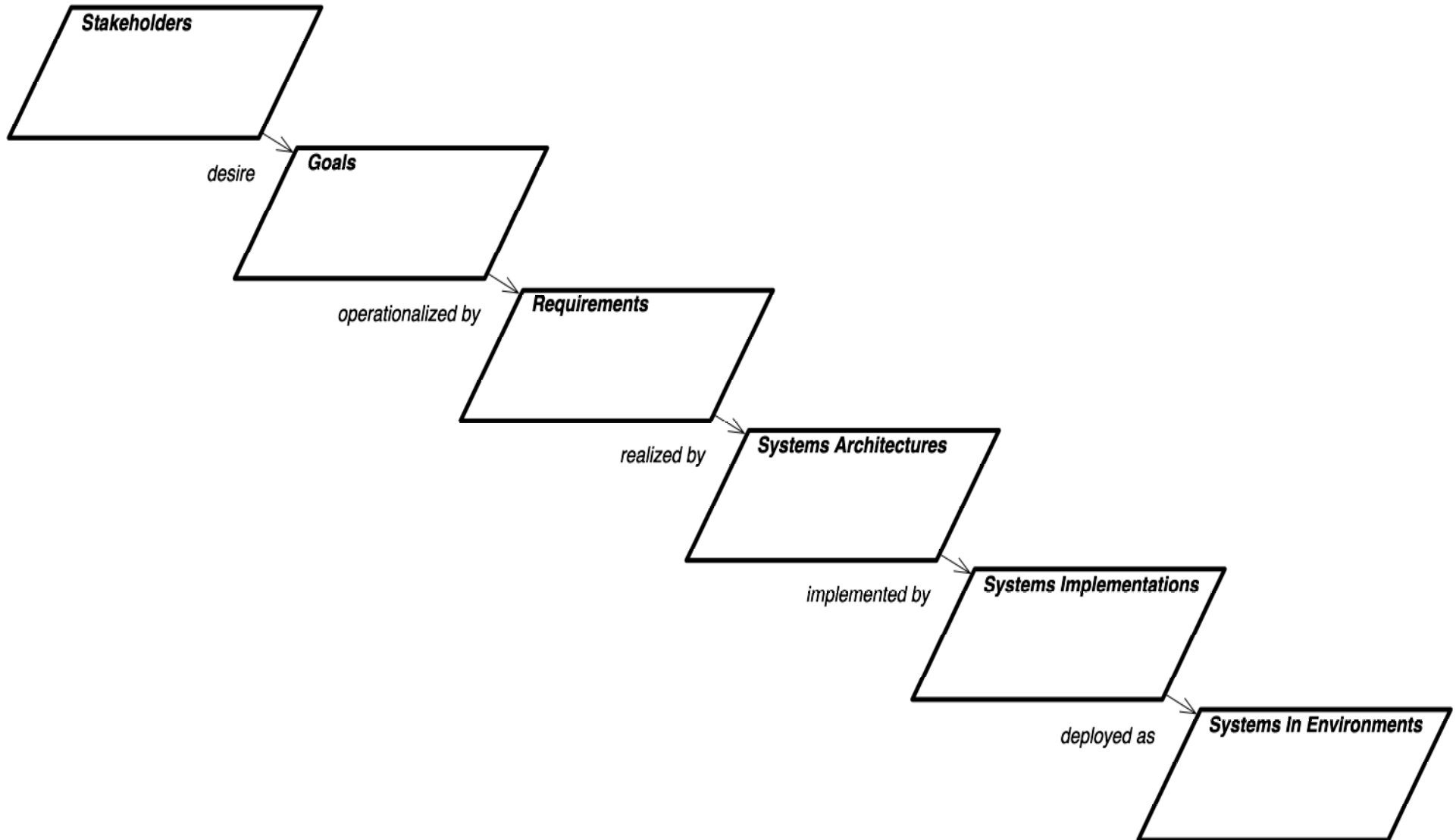 - ❖ It plugs some of the holes in existing models and methods

Deployed Systems (of Systems)

Systems In Environments

Hadar Ziv / ziv@ics.uci.edu

Stakeholders use Deployed Systems

Stakeholder-to-System Lifecycle

Stakeholders

desire

Goals

operationalized by

Requirements

realized by

Systems Architectures

implemented by

Systems Implementations

deployed as

Systems In Environments

Hadar Ziv / ziv@ics.uci.edu

Artifact Details at Lifecycle Layers

Hadar Ziv / ziv@ics.uci.edu
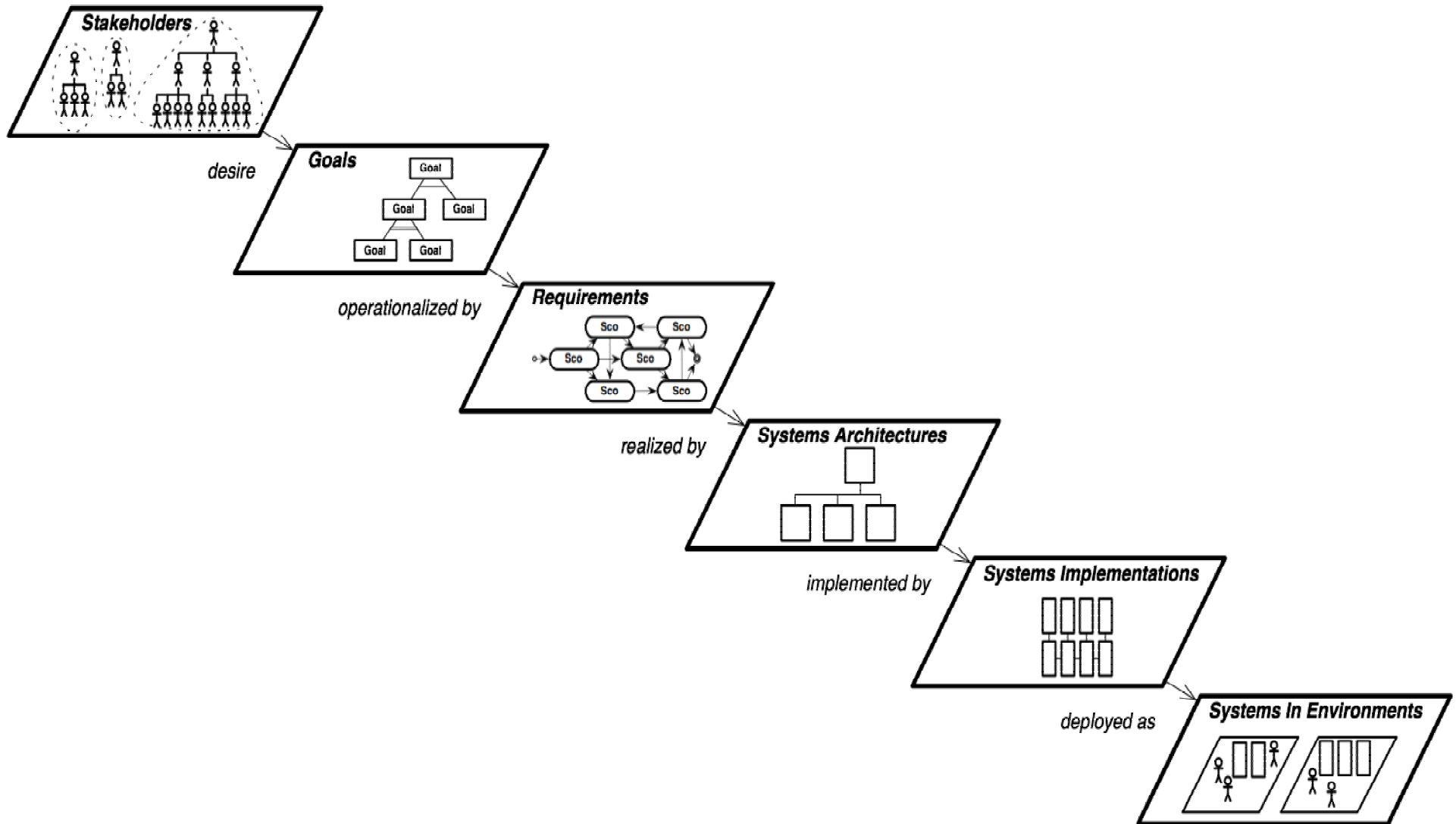
Add Testing to Lifecycle

Stakeholders

↓ *desire*

Goals

↓ *operationalized by*

Requirements

↓ *realized by*

Systems Architectures — *derive* → **System/Integration Tests**

↓ *implemented by*

Systems Implementations ← *validate* — **System/Integration Tests**

Systems Implementations — *derive* → **Component/Unit Tests**

Systems Implementations ← *validate* — **Component/Unit Tests**

↓ *deployed as*

Systems In Environments

Acceptance Tests — *validate* → **Systems In Environments**

Operational Tests — *validate* → **Systems In Environments**

derive (Goals → Acceptance Tests)

derive (Requirements → Operational Tests)

Some Challenges/Opportunities

- Stakeholders and Goals modeled explicitly
  - Stakeholder goal analysis
  - Acceptance Testing derived from Goals
- Requirements modeled explicitly
  - Traceability between Goals and Requirements
  - Will this change how we model Requirements?
    - » Scenarios
  - Operational Testing derived from Requirements
    - » Scenario-based Testing
- Deployed systems modeled explicitly
  - Testing of deployed systems
    - » Residual Testing

Highlights and Future Work

- A complete lifecycle should not ignore
  - Stakeholders, goals, goal networks, scenarios, scenario networks, episodes, concerns, aspects
  - Testing of the above, Traceability among all of the above
- We are doing (or wanting to do) work in
  - goal analysis and goal decomposition
  - Scenario composition, scenario refactoring, scenario analysis (including episodes)
  - Design and implementation using concerns and aspects (for cross-cutting concerns)
  - Scenario-based testing, aspect testing, residual testing
- Possible cooperation and collaboration with Aerospace
  - Applying and validating ideas
  - Templates/frameworks for goals/scenarios

Hadar Ziv / ziv@ics.uci.edu