A Taxonomy of Requirements Aspects

Nan Niu\(^1\), Steve Easterbrook\(^1\), and Yijun Yu\(^2\)

\(^1\)Department of Computer Science
University of Toronto, Canada
Email: \{nn, sme\}@cs.toronto.edu

\(^2\)Computing Department
The Open University, UK
Email: y.yu@open.ac.uk

March 13, 2007
EA & RE: Strong Connection

  ⇒ One of the earliest descriptions of early aspects

  ⇒ Best paper award in RE’06
Many Forms of Requirements Aspects

- A collaboration in requirements for components [Grundy’99]
- An extension in a use case diagram [Jacobson’03]
- A softgoal in a goal model [Yu’04]
- An instance of terminological interference in viewpoints-based requirements models [Niu’07]
- A non-functional requirement in a software requirements specification [Cleland-Huang’06]
- ...
A Feature Diagram for Requirements Aspects

Requirements Aspect

Intent (Purpose)
  - Observable Behavior
    - Use Case
    - Goal
    - ViewPoint
Form (Author's Chosen Organization)
  - (Structured) Natural Language
Weaving Mechanism
  - refer to Table 1 for detailed discussion
Kind
  - Functional (Technical)
    - e.g., logging, fault tolerance, buffering, etc.
  - Non-Functional (Quality)
    - e.g., security, availability, usability, etc.
Life Cycle
  - Source
  - Target
  - Refinement
    - Mapping
    - Influence
    - Base
    - Aspect
Interpretations from an RE Perspective

- **Metaphor**: Every requirements aspect acts as a service provider to some base modules.
- **Advice** defines the body or the content of the service. It describes *what* the service is about.
- **Join points** are points in the base which a requirements aspect interacts with. They describe *where* the service is provided.
- **Pointcut** represents a set of join points. It describes the *situational patterns* of the service.
- **Weaving** is the process of coordinating service providers (requirements aspects) and consumers (base requirements). It describes *when* and *how* the service takes place.
- **Intent** describes *why* the service is needed in the first place.
What else is in the paper?

- Crosscutting properties
  - constraints, dependencies, trade-offs, rationale, consistency, etc.
- Comparison and classification of requirements aspects approaches
- A reified requirements aspect – security
  - Goal models
  - Problem frames
  - (Mis)use cases
- Related work
Concluding Remarks

⇒ A preliminary attempt to characterize features of a requirements aspect as it is used today in the literature.

⇒ Should it be modified, simplified, or extended with additional features? Can it be generalized to cope with symmetric approaches, and to handle architectural-, code-, and test-level aspects?

⇒ The time is ripe for consolidating early aspects knowledge into a set of polished best practices and patterns.