



**Carnegie Mellon
Software Engineering Institute**

The Business Case for Requirements Engineering

RE'2003

12 September 2003

**Donald Firesmith
Acquisition Improvement Team
Acquisition Support Program
Software Engineering Institute (SEI)
Carnegie Mellon University
Pittsburgh, PA 15213**



In a Nut Shell

- **Requirements first opportunity to screw up**
- **Many requirements engineers aren't**
- **Requirements typically contain many defects**
- **Requirements impact all down-stream work**
- **Cost to fix defects increases rapidly the earlier they are introduced**
- **Requirements primary reason for failure**



First Opportunity to Fail

There are many chances to fail on any project:

- **Contracting**
- **Requirements Engineering**
- **Architecting**
- **Design**
- **Implementation**
- **Integration**
- **Testing**
- **Etc.**

Requirements first engineering chance to fail.



Many Requirements Engineers Aren't

Requirement Myth:

- Since most requirements are specified in narrative English and most employees are minimally literate, managers often think that anyone (including low-level new hires) can do requirements engineering.

Requirements engineers lack training in:

- Requirements Tasks:
 - Requirements Identification
 - Requirements Analysis
 - Requirements Specification
 - Requirements Management
- Requirements Techniques (e.g., use case modeling)
- Requirements Tools



Requirements Contain Defects

The percentage of defects originating during requirements engineering are estimated as:

- 50% (Karl Wieggers, 2001)
- 42% (A Wingrove)
- 60-64% (requirements and design – EBG Consulting)

Requirements typically lack:

- Cohesiveness, Completeness, Correctness, Consistency, Currency, Essential, Feasibility, Lack of Ambiguity, Relevance, Testability, Usability, Validatability



Requirements Engineering Impacts

Requirements Engineering impacts:

- Management (scope management)
- Architecture (architecturally-significant requirements)
- Design and Implementation
- Testing
- Quality Engineering (determines defects)
- Safety Engineering (safety requirements)
- Security Engineering (security requirements)
- Reuse
- Training

Requirements Defects Snowball



Defect Costs Are Excessive

Requirements engineering defects cost:

- 50-200 times as much to correct once fielded. (Barry Boehm, 1988)
- 10-100 times as much to correct once fielded (Steve McConnell, 2001)
- 15 times as much to correct once fielded (IBM System Sciences Institute – all defects so requirements worse)
- 10 times as much to correct during testing (Hughes Aircraft)

Reworking requirements defects on most software development projects cost:

- 40-50% of the effort (Capers Jones)
- 80% of the effort (Karl Wieggers, 2001)



Bad Requirements Cause Failures

Requirements problems are the single number one cause of project failure:

- **Significantly over budget**
- **Significantly past schedule**
- **Significantly reduced scope**
- **Poor quality applications**
- **Not significantly used once delivered**
- **Cancelled**



Conclusion

Requirements Engineering:

- **Starts project on the right foot**
- **Turns requirements workers into trained requirements engineers**
- **Eliminates and minimizes defects**
- **Improves architecting, design, implementation, testing, QA, security, safety, etc.**
- **Decreases development and lifecycle costs**
- **Increases probability of success**



Carnegie Mellon
Software Engineering Institute

Contact Information

Donald Firesmith

Senior Member of the Technical Staff
Acquisition Improvement Team
Acquisition Support Program
Telephone: 412-268-6874
Email: dgf@sei.cmu.edu

U.S. Mail:

Software Engineering Institute (SEI)
Carnegie Mellon University
Pittsburgh, PA 15213-3890

World Wide Web:

<http://www.sei.cmu.edu/>
<http://www.donald-firesmith.com/>