How To Define “Lean and Mean” Requirements

NDIA CMMI Conference
November 2008

Timothy G. Olson, President
Lean Solutions Institute, Inc.
(760) 804-1405
Tim.Olson@lsi-inc.com
www.lsi-inc.com

“I have made this letter longer than usual because I lack the time to make it shorter”

Blaise Pascal
Objectives

Describe some requirements problems from industry.

Present a useful classification of requirements problems.

Describe some practical strategies and best practices to successfully define “lean requirements” that address the requirement problems.

Provide real examples that address requirements problems.

Answer any of your questions.

Outline

Why Focus on Requirements?

A Practical Requirements Classification

Lean Overview

Lean Approaches for Requirements

Lean Requirement Examples

Summary
Why Focus on Requirements?

The hardest single part of building a system is deciding what to build...
No other part of the work so cripples the resulting system if done wrong. No other part is more difficult to rectify later.

Adapted from Fredrick Brooks, Jar. [Brooks 87]

Why Focus on Requirements?

A research report from the Standish Group highlighted the continuing quality and delivery problems in our industry and identified three leading causes:

• Lack of user input
• Incomplete requirements and specifications
• Changing requirement specifications

Outline

Why Focus on Requirements?

A Practical Requirements Classification

Lean Overview

Lean Approaches for Requirements

Lean Requirement Examples

Summary

Problems with Requirements

According to the SEI [Christel 92], problems of requirements elicitation can be grouped into 3 categories:

1. Problems of Scope: the requirements may address too little or too much information.

2. Problems of Understanding: problems within groups as well as between groups such as users and developers.

3. Problems of Volatility: the changing nature of requirements.
Scope and Volatility

The list of 10 requirements elicitation problems given in [McDermid 89] can be classified according to the 3 categories in [Christel 92]:

Problems of Scope

• The boundary of the system is ill-defined
• Unnecessary design information may be given

Problems of Volatility

• Requirements evolve over time

Problems of Understanding

• Users have incomplete understanding of their needs
• Users have poor understanding of computer capabilities and limitations
• Analysts have poor knowledge of problem domain
• User and analyst speak different languages
• Ease of omitting “obvious” information
• Conflicting views of different users
• Requirements are often vague and untestable, e.g., “user friendly” and “robust”
Outline

Why Focus on Requirements?
A Practical Requirements Classification

Lean Overview

Lean Approaches for Requirements
Lean Requirement Examples
Summary

What is Lean?

Lean has its roots in quality and manufacturing, and is a recent popular movement in quality.

“Lean Production” is the name for the Toyota Lean Production System.

The following are major lean references (see references in back of presentation for full references):
• “The Machine That Changed The World”
• “Learning to See”
• “The Toyota Way”
• “The Toyota Product Development System”
• “Lean Thinking”
Some Lean Principles - (1)

Establish customer defined value (i.e., identify the “value stream”). Process = “value”.

Continuously eliminate non-value added activities (e.g., waste, rework, defects).

Use leadership and standardization to create a lean culture.

Align your organization through visual communication.

Create an optimized process flow (e.g., “Flow”, “Pull”, “Just-In-Time”, “Leveled”).

Some Lean Principles - (2)

Use lean metrics to manage the value stream.

Front-Load the process for maximum design space.

Build a learning organization to achieve lean and continuous improvement.

Adapt technology to fit your people and processes.

Strive for perfection through continuous improvement.
Quality Maturity

<table>
<thead>
<tr>
<th>STAGE</th>
<th>SUMMARY</th>
<th>COQ</th>
<th>BA</th>
<th>DCF</th>
<th>SEI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prevention</td>
<td>“We know why we have happy customers.”</td>
<td>5%</td>
<td>800</td>
<td>20%</td>
<td>5</td>
</tr>
<tr>
<td>Wellness</td>
<td>“Quality planning, control, and improvement are routine.”</td>
<td>10%</td>
<td>700</td>
<td>40%</td>
<td>4</td>
</tr>
<tr>
<td>Progressive Care</td>
<td>“Management commitment and continuous improvement resolve quality problems.”</td>
<td>18%</td>
<td>600</td>
<td>60%</td>
<td>3</td>
</tr>
<tr>
<td>Intensive Care</td>
<td>“We don’t know why we have quality problems, but they hurt.”</td>
<td>25%</td>
<td>400</td>
<td>80%</td>
<td>2</td>
</tr>
<tr>
<td>Comatose</td>
<td>“What quality problems?”</td>
<td>33%</td>
<td>200</td>
<td>100%</td>
<td>1</td>
</tr>
</tbody>
</table>

• Acronyms are (COQ=Cost of Quality; BA=Baldrige Award; DCF=Dilbert Correlation Factor; SEI=SEI CMMI/CMM)

• Based on “The Eternally Successful Organization”, by Crosby, the SEI, the Baldrige Award, & Dilbert Comics

Outline

<table>
<thead>
<tr>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Why Focus on Requirements?</td>
</tr>
<tr>
<td>A Practical Requirements Classification</td>
</tr>
<tr>
<td>Lean Overview</td>
</tr>
<tr>
<td>Lean Approaches for Requirements</td>
</tr>
<tr>
<td>Lean Requirement Examples</td>
</tr>
<tr>
<td>Summary</td>
</tr>
</tbody>
</table>
Lean Requirements Strategies

1. Write lean requirements.
2. Use operational definitions to define “good” lean requirements.
3. Use a lean Requirements Processes.
4. Use lean Configuration Management (CM) and CM Metrics.
5. Use lean requirement metrics.
6. Use a lean requirements standard.
7. Use lean early defect detection and defect prevention.

Outline

Why Focus on Requirements?
A Practical Requirements Classification
Lean Overview
Lean Approaches for Requirements

Lean Requirement Examples

Summary
1. How To Write Lean Requirements

“Chunk” requirements (e.g., 7 ± 2) into products, product components, and usage scenarios.

Use architectures and models to help select the best “chunks” (also helps to reuse requirements).

Write 1 sentence lean requirements (can have 1 sentence sub-requirements), use an operational definition of a lean “good requirement”, and think of requirements as a record (e.g., DB, tool) with attributes (e.g., source, metrics, traceability, etc).

Use a requirements writing checklist, for example:
- Question every requirement: “Value added”?
- Question every requirement attribute: “Value added”?
- Question every word of every requirement
- Requirement Measurable? Testable? Traceable?
- Have Functional Requirements? Performance Interface?

2. Example Operational Definition

What is a good requirement? A lean requirement? When is a requirement defined? Questions like these are very difficult to answer without an operational definition.

An operational definition precisely and concisely defines a measurable requirement that states [Adapted from NASA 96]:
- What does the requirement have to do? (in 1 sentence)
- How well? (e.g., ± limits, quality, in measurable terms)
- Under what conditions? (e.g., environment, states)
2. Example Lean Requirements

<table>
<thead>
<tr>
<th>#</th>
<th>Requirement (What)</th>
<th>Conditions</th>
<th>Upper Limit</th>
<th>Lower limit</th>
<th>Base Measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Report total percentage of students that passed the first test and graduated</td>
<td>Students that pass first test by =&gt; 70% score</td>
<td>Calculate Percentage to 3 decimal places</td>
<td>Plus or minus .001</td>
<td>Percent</td>
</tr>
<tr>
<td>2</td>
<td>Report total percentage of students that failed the second test and did not graduate</td>
<td>Students that failed second test by &lt; a 70% score</td>
<td>Calculate Percentage to 3 decimal places</td>
<td>Plus or minus .001</td>
<td>Percent</td>
</tr>
</tbody>
</table>

3. Lean Doc. Framework

- **Policies**
  - "Laws" or "Principles" that govern operations
  - Lean Requirements Policy - 1 Page

- **Standards**
  - "Operational definitions" & "acceptance criteria"
  - Lean Requirements Standard - 1 page
  - Lean Process Models:
    - 5 W's in Diagram - 1 Page
    - Supporting Text - 1 Page
  - Lean Requirements Checklists - 1 Page

- **Processes**
  - "What happens over time" to build products
  - Implemented by

- **Procedures**
  - "How to" or step by step instructions

- **Training**
  - Provides the needed knowledge and skills

- **Tools**
  - Supports and automates operations

- **Honkin' Req. Binder - OK**

- Supported by

- 1 Record Per Requirement

- Slide adapted from "A Software Process Framework for the SEI Capability Maturity Model", CMU/SEI-94-HB-01
3. Define Lean Requirements Processes (REQM, RD, CM)

- Customer Requirements
- Develop Requirements (RD)
- Product Requirements
- Manage Requirements (REQM, CM)
- Updated Customer Requirements
- Updated Product Requirements
- Change Requests
- Problem Reports

3. Requirements Process - NASA Onboard Shuttle Project

- Requirements conception
  - Identify need
  - Examine architectural options
  - Develop software system solution
- Requirements generation
  - Define software requirements in accordance with operational concept and system requirements
  - Produce requirements specification
- Requirements analysis
  - Assess technical and resource impact
  - Determine acceptability, implementability, testability
  - Examine requirements readiness
- Requirements inspection
  - Discuss proposed requirement in detail
  - Discuss operational scenarios
  - Identify issues and errors
- Requirements approval
  - Evaluate risks and benefits
  - Decide on resource expenditures
  - Establish baseline
- iteration
  - Correct errors
  - Resolve issues
  - Rewrite
4. Example Lean NASA JPL MGSS CM Process

- 7.1 Perform CM Planning
- 7.2 Perform Configuration Control
- 7.3 Perform Configuration Status Accounting
- 7.4 Perform CM Audits


4. Use CM and CM Metrics

Fundamental Baselines

- Requirements Baseline → Implementation Baseline → Product Baseline

Place the requirements under formal CM and use CCB’s to control changes.

Example CM Metrics:
- Number of CRs/PRs (e.g., open vs. closed over time)
- Requirements Volatility (e.g., number of CRs per requirement)
5. Example Lean Metrics

<table>
<thead>
<tr>
<th>#</th>
<th>Requirement</th>
<th>Reference (e.g., customer)</th>
<th>Allocation</th>
<th>Stability (H/M/L)</th>
<th>Risk (H/M/L)</th>
<th>Priority (H/M/L)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>System shall send an RTF FAX</td>
<td>SOW # 10-20.3</td>
<td>Software</td>
<td>H</td>
<td>L</td>
<td>M</td>
</tr>
<tr>
<td>2</td>
<td>Aircraft position shall be updated by the Inertial Navigation System (INS) Solution</td>
<td>ORD #2-30-20.3.4.4</td>
<td>Software</td>
<td>M</td>
<td>M</td>
<td>H</td>
</tr>
</tbody>
</table>

6. IEEE SyRS and SRS

**SyRS**

1.0 Introduction  
2.0 General System Description  
3.0 System Capabilities, Conditions, and Constraints  
   3.1 Physical  
   3.2 System Performance Characteristics  
   3.3 System Security  
   3.4 Information Management  
   3.5 System Operations  
   3.6 Policy and Regulation  
   3.7 System Life Cycle  
4.0 System Interfaces

**SRS**

1.0 Introduction  
2.0 Overall Description  
3.0 Specific Requirements  
   3.1 External Interface Requirements  
   3.2 Functional Requirements  
   3.3 Performance Requirements  
   3.4 Design Constraints  
   3.5 Software System Attributes  
   3.6 Other Requirements  
Appendices  
Index

* Adapted from: IEEE Std 1233-1998  
* Adapted from: IEEE Std 830-1998
6. Organizing SRS Section 3

SRS Section 3 can be organized by:

- Mode
- User Class
- Object
- Feature
- Stimulus/Response
- Functional Hierarchy
- Multiple organizations

7. Example Requirements Checklist Categories

1. Clarity
2. Completeness
3. Complexity
4. Consistency
5. Constraints
6. Feasibility
7. Functionality/Logic
8. Interfaces
9. Standards
10. TBDs
11. Testability
12. Traceability
Etc.
Summary

The hardest single part of building a system is the requirements.

The top requirements problems are inadequate requirements specifications, changes to requirements, and lack of user input.

Lean is a very powerful approach to improve the quality, productivity, and performance of requirements, systems engineering, and software engineering.

There are lean strategies that you can use today that will help you address problems with requirements.
How To Define
“Lean and Mean”
Requirements

NDIA CMMI Conference
November 2008

Timothy G. Olson, President
Lean Solutions Institute, Inc.
(760) 804-1405
Tim.Olson@lsi-inc.com
www.lsi-inc.com


Training Material Used with Permission and Licensed by Lean Solutions Institute, Inc. (LSI)