Requirements Engineering – Industry Practice

Requirements Engineering is the disciplined and systematic approach to elicit, specify, analyze, commit, validate and manage requirements while considering user, technical, economic and business-oriented needs and objectives. It is an engineering discipline because of this disciplined and systematic approach. The goal of Requirements Engineering (abbreviated as RE) is to develop good – not perfect – requirements and to manage them during development with respect to risks and quality. RE is the discipline within systems and software engineering that bridges the entire life cycle and thus determines success or failure of a product or project. In short, it is what makes the difference between a winning product and a mere set of features.

RE is both a problem-oriented and a solution-oriented discipline. As a problem-oriented discipline, RE interfaces with product management, product design and systems engineering in that it analyzes the market and client challenges that exist in the system in which the product is embedded. Problem-oriented RE borrows from product management and psychology; it deals with goals to be achieved, the stakeholders who have these goals, and the problems to be solved within given business constraints. As a solution-oriented discipline, RE interfaces with product development in that it specifies the desired functions, quality attributes, and other properties of the software that is to be built or assembled. Because both views of RE are valid, RE is a discipline that maps needs to solutions. We have seen both perspectives in many OOP presentations, and we continuously face discussions across industries and research, when new techniques show up for the first time, such as service-orientation, use cases, or new elicitation techniques.

Requirements Engineering has evolved across industries over the past years. Here are some of the major RE success factors which apply to all sorts of projects, products and services:

- **Requirements Engineering is team work**: Winning products depend on collaboration between marketing, sales, product management, and software development.
- **Don’t start with technical requirements**: A product should create an experience, a value for its user. Look to the major use cases from a user perspective and then distill requirements and priorities.
- **Do not get lost in feature battles**: Remember that your product is what the consumer perceives, and not what you sell. Use lean and agile techniques to reduce complexity and to focus on what matters.
- **Systematically analyze requirements**: Look to your estimates, impact analyses and planning before the project, at key milestones, and when there are changes to scope or content.
- **Manage risks**: Have a profound understanding of your position and product strategy. Periodically
evaluate products and markets to avoid dead-ends and unnecessary features.

- **Work professionally**: Follow a suitable process for developing and managing your requirements. Use templates, specify and document agreements, and work systematically. Avoid ad-hoc changes because they mean rework.

Current trends are making effective requirements practices even more important.

- **Value creation with customers**. Value-oriented requirements engineering will grow rapidly, i.e. improving the evaluation of requirements within a business case from a portfolio management perspective. Customers are part of the value creation. This implies dynamic segmentation down to the single-buyer segment. Questions include: What is the customer business case behind the requirements? Is the own business case valid and in line with the customer or user business case? What is the contribution of requirements to this business case?

- **Innovative market rules and business models**. Many industries today have such low entry levels, that a new competitor is literally a mouse-click away. Friction-free deliveries further add to this competitive trap. It is therefore important to develop needs and related requirements continuously from scratch and innovate how their current and prospective customers can further improve their own business. Examples include community networks, wikinomics with a global and open access to information and resources, and blue ocean approaches to create new value for customers rather than fighting on price alone.

- **Agility to manage uncertainties**. Better predict changes to requirements on an individual level. Which requirements are most volatile and at same time exposing the project to highest risk? How can they be addressed by sufficiently flexible solution architecture? Prioritize requirements according to needs and value creation. Use agile and lean processes to systematically implement according to priorities.

- **Service matters**. Requirements go far beyond those of the technical features and components. They include service, evolution, integration of business processes and the like. It is about speed to needs. RE has to cope with this need and not get cornered by only looking towards what are mere software features. In the end the customer is not interested in features, but in his needs and how they are best satisfied by products and services.

- **Quality focus**. Quality starts at the start of a product concept and a project. Quality needs are manifold and can contradict each other. RE has to deal with these constraints and must offer win-win solutions. Further develop a quality perspective in RE, covering for instance the usage of commercial components, including a variety of partners or suppliers, managing the quality delivered by such external partners, adapting systems quality requirements as business needs change.

- **Supplier networks and eco-systems**. Aside from the classic competitive schemes, customers increasingly demand collaboration and sustainable networks of suppliers. RE is not anymore a concentrated activity, but means intense and extensive collaboration. The traditional concept of supply-chain will be replaced by adaptive supply networks. Suppliers are even more than ever subject to continuous evaluation and replacement where necessary. The success of a supplier depends how well he is able to create communities and business models together with customers and other suppliers.

- **Effective knowledge management**. To remain competitive, reuse of requirements and solution components will further grow. Introduce knowledge management techniques for simulative collection, evaluation, modeling and retrieval of requirements and underlying decisions. Use appropriate tools to manage requirements along with derived artifacts, such as architecture, models, code segments, test cases, etc.

Requirements engineering will further grow in its impact. Taken together, these challenges provide an exciting context for requirements engineering in the context of the next twenty years of OOP.

For more information on requirements engineering we recommend following internet resources:


**Author**

**Dr. Christof Ebert** is managing director at Vector Consulting Services. He supports clients around the world to improve product strategy and product development and to manage organizational changes. Prior to that, he held global management positions for ten years at Alcatel-Lucent. A trusted advisor for companies around the world, member of industry boards, he lectures at the Universities of Stuttgart and at the Sorbonne in Paris. He authored several books including his most recent book “Global Software and IT” published by Wiley. He received the IEEE distinguished visitor award and is a member of the Alcatel Technical Academy. Contact him at christof.ebert@vector.com

---

Instructor: Dr. Christof Ebert

Dr. Christof Ebert is managing director at Vector Consulting Services.

He supports clients around the world to improve product strategy and product development and to manage organizational changes. Prior to that, he held global management positions for fifteen years at Alcatel-Lucent.

Dr. Ebert serves on a number of advisory and industry bodies. An internationally renowned keynote speaker, he teaches at the University of Stuttgart and authored several books including his most recent book “Global Software and IT” published by Wiley.

christof.ebert@vector.com
www.vector.com/consulting

Requirements Engineering
Dr. Christof Ebert

Vector Consulting Services

- offers a comprehensive consulting portfolio for optimizing product strategy and development
- with clients such as Accenture, Audi, BMW, Bombardier, Bosch, Daimler, Hyundai, IBM, Lufthansa, Munich RE, Siemens, Telefonica, Thales, Valeo, Vodafone, Zeiss
- as a group serves companies across the world with over 1200 employees and well over 200 Mio € pa
- www.vector.com/consulting

Agenda

- Snapshot on Requirements Engineering
- Industry Challenges
- Applying Requirements Engineering
- Delivering Results

Disclaimer:
Worldwide copyright of pictures is with dpunkt publishing group and/or with Vector.
Contact author before copying or using content.
Challenge „Software Intensive Systems“

Major revenue source – and high business risk...

- Missing transparency
- Delays and cost overruns
- Insufficient productivity
- Rework and defects
- Project and liability risks
- Demotivation and burn-out

... and no time to improve

![Success rate of projects](image)

- 18% Successful
- 39% Late or over budget
- 43% Cancelled

Primary sources for project failure

<table>
<thead>
<tr>
<th>Insufficient processes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unclear responsibilities</td>
</tr>
<tr>
<td>Inadequate requirements</td>
</tr>
</tbody>
</table>

Sources
Top: Standish Group 2012, ca. 10000 projects
Bottom: Standish Group, ObjectGroup, McKinsey, 2013

Insufficient Requirements Have a Long Tradition ...

And suddenly there was an intense dispute between the king and his supplier ...

But, this time the king would not apply the typical punishments but rather listen to his advisors. He understood that he is part of the problem.

He understood that a sustainable solution needs change on both sides.

Requirements Engineering was introduced.

There are Three Requirements Related Risks

1 Missing Requirements
- Customer not involved
- Critical needs overlooked

2 Wrong Requirements
- Mixing “what” (needs) and “how” (solution)
- Inadequate verification

3 Changing Requirements
- Unknown baseline
- Insufficient change management


Reality: Industry Case Study from a Client Project

- Vague requirements
- Insufficient analysis and specification
- Changing requirements

- Unrealistic plans and commitments
- Burn out: Decreasing motivation
- Schedule pressure
- Defects: insufficient verification
- Schedule delay
- Poor quality

- Unmet customer expectations
- Lost future business
Break and Think

What are your business challenges?
How are they related to Requirements Engineering?
Summarize your challenges to obtain the right stimulus from this training.

... 
... 
... 

One Requirement – And Many Questions

REQ_0815
The floor console allows calling the elevator to individual floors. The console consists of two buttons to select the directions up and down. The user calls the elevator by pressing the call button. The VIP function is activated by a key switch. This allows to directly access floors.

- Does the requirement deliver value?
- Are contents relevant or just a feature list?
- Is the requirement feasible?
- What is the status of the requirement?
- Who is responsible for this requirement?
- How will the requirement be validated?
- How will the requirement been implemented?

What is Value?

- Formal: The extent to which a good or service is perceived by its customer to meet his needs, measured by customer’s willingness to pay for it.
- Can be a combination of systems, solutions, materials and services delivered as is or as a component for another product

Example: R&D Contribution to Value Generation

iPhone: mobile phone with reliable technology, appealing design and a secured supply chain for applications, content and payment.

R&D Value generation:

- R&D innovation only for few key differentiators, such as display, camera. Profits are immediately used to further innovate and thus generate new needs in short timeframes.
- Reliable technology. Innovate, where it matters. Design to Cost. It is not primarily about new technology but about improving of what is already around.
Requirements

**Requirement**
- A condition or capability needed to solve a problem or achieve an objective
- Different types of requirements; recursively applicable

Requirements Engineering

**Requirements Engineering**
- The disciplined and systematic approach (→ "engineering") to:
  - elicit, document, analyze, agree, verify and manage requirements
  - while considering value-oriented, technical and business-driven needs.

Requirements Engineering and its Context

**Value orientation:**
Ca. 50% of all functions in a product are not used; only 20% of all functions actually deliver value

**Early defect removal:**
80% of all defects detected in test and 43% of all defects in field result from insufficient RE.

**Cost reduction:**
Typically 3-6% of effort goes into RE. Doubling this effort reduces life-cycle cost by 20-40%.

Benefits of Requirements Engineering

Requirements Engineering is a key industry success factor.

Sources: Ebert, 2010; Standish Group, 2009; Institut für angewandte Informatik Karlsruhe, 2005; MGI Business Analysis Benchmark 2008 (103 companies, avg. 3 Mio US$ project volume)
Example: Cost Reduction with Requirements Engineering

- Early defect removal
- Better project preparation
- Less requirements changes
- Uncertainties and risks are better managed
- Better project control

Good requirements engineering reduces project cost.


Example: Better Efficiency with Requirements Engineering

<table>
<thead>
<tr>
<th>Typical causes for delays</th>
<th>Four techniques had been selected to improve</th>
<th>Implementing all four techniques dramatically improved schedule</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conflict of interest; agreements are not kept</td>
<td>Core team with all relevant stakeholders</td>
<td>Actual schedule adherence (100% = Agreed date is kept)</td>
</tr>
<tr>
<td>Unclear business case of requirements</td>
<td>Product lifecycle with mandatory milestones</td>
<td>Goal: 105-110%</td>
</tr>
<tr>
<td>Incomplete requirements</td>
<td>Requirements are analyzed and prioritized</td>
<td>Amount of the four techniques implemented simultaneously</td>
</tr>
<tr>
<td>Dependencies are overlooked</td>
<td>Dependable portfolio management</td>
<td></td>
</tr>
</tbody>
</table>

Source: Alcatel-Lucent 2006

Break and Think

Where in the life-cycle do you see most value from professional requirements engineering?

Give concrete industry examples about the levers and the value.

- ...
- ...
- ...

Agenda

- Snapshot on Requirements Engineering
- Industry Challenges
- Applying Requirements Engineering
- Delivering Results
Industry Trends: Critical Systems

- **Safety-critical**: Critical for health and sustainability
  - Example: Loss of lives, injuries, environmental damages
- **Security-critical**: Critical for preserving information integrity
  - Example: Sensitive data is accessed thus impacting safety
- **Business-critical**: Critical for business success
  - Example: Loss of business, damage to image

Critical Systems: Characteristics and Challenges

- **Ubiquity**
  Software failures have great potential to affect people and societies. Software is critical even when it is not complex.

- **Complexity**
  Embedded software is connected with other systems. Multiple eco- and supply-systems are entangled to deliver services.

- **Value**
  Users and society expect more from systems. Functionality and evolution cycles will further accelerate and evolve.

- **People**
  Software is embedded to systems in which people participate. Human interactions increasingly puts systems and people at risk.

Characteristics and challenges are similar across industry domains. It is worth learning from other industries and cross-fertilizing.

Challenge: Ubiquity

- **The embedded market values 200 B€ with 10% growth rate** (Automotive: ca. 15% CAGR)

Demand: Manage consistency and interdependencies

**Sources**: Jones/Ebert 2009, Vector 2013
Vector Client Challenges in 2014

**Efficiency**
- Global competition demands 10-20% cost reduction for same output on annual basis.
- Customers expect intelligent efficiency improvement (cost, cycle time).

**Innovation**
- Building on existing products, Green fields are rare. Seamless innovation.
- Customers expect iterative innovation cycles with clear ROI.

**Quality**
- Quality matters. Painful experiences across industries enforce safety, security, reliability.
- Customers expect lean processes to deliver and to manage risks.

**Flexibility**
- R&D delivered by worldwide supplier networks growing from 25 to 75% share.
- Customers expect support on more flexible business models.

**Technology**
- Exploding complexity. Technology and strategy need to be controlled by business needs.
- Customers expect adaptive road-mapping and variant management.

---

Your Own Requirements Engineering ...

Think about your own work and current projects...

- Are requirements consistently developed?
- Are requirements systematically addressed?
- Are relevant business needs understood and considered?
- Are requirements changes handled in a disciplined way?
- Does it all depend on management?

---

Break and Think

Think about your own work and current projects...
Rate your own experiences from 0% - 100%

- Are all relevant requirements systematically addressed?
- Are market and business needs understood?
- Are requirements changes handled in a disciplined way?
- Does your management stimulate systematic requirements engineering?

---

Agenda

- Snapshot on Requirements Engineering
- Industry Challenges
- Applying Requirements Engineering
- Delivering Results
Four Bold Requirements Engineering Themes

- Value innovation
- Product management
- Collaboration
- Quality requirements

Product Management – Challenge

More than half of all delivered features are not perceived as value by customers – and will create more life-cycle cost than benefits. 40-50% of the R&D budgets are mis-investments due to too much variability, wrong functionality, rework and waste.

- Rarely 19%
- Sometimes 16%
- Often 13%
- Always 7%
- Never 45%

Sources: Microsoft, Daimler, BITKOM 2012, Ebert 2013

Product Management – Solution

Vector recommendations:
- Establish a core team with all relevant functions
- Empower the product manager to drive the product as „entrepreneur”
- Manage variation with roadmaps, product line engineering, planned reuse
- Apply target costing and consistent valuation
- Use priorities to manage uncertainty, change and planning errors
Product Management – Benefits

Impact (with normalized starting point at 100%)


Vector project experiences:
- Accelerated market entry by 20-30%
- Reduced defects and delays by up to 80%
- Improved customer satisfaction by 20%

Product management deployment phases

Quality Requirements – Challenge

Problems with radiation system:
- Interception and overload allows glitches to supervisory system
- American healthcare supplier, 2012

Problems with acceleration:
- Car unintentionally accelerates thus causing personal damage
- Japanese car maker, 2013

Problem with traffic management:
- Operator interface is unintentionally inactivated
- European systems supplier, 2011

Problems with safety control:
- Person diagnosis are not or too late activated
- UK robotics company, 2013

The risk of damage due to insufficient handling of safety requirements is high (injuries, reliability, safety, costs, damage to reputation, etc.)

Quality Requirements – Solution

Vector recommendations:
- Systematically elicit and prioritize quality requirements and relate to business needs
- Use quality gates as defined thresholds and to ensure transparent governance.
- Use consistent process, DIA, project schedule and manual for Safety Plan
- Trace quality requirements to architecture, design decisions, test strategy
Quality Requirements – Benefits

Vector project experiences:
- For low to high reliability and safety levels, relative cost can be cut by 10-20% with improved process maturity.
- Cost savings are achieved by early defect detection, focused test, automated test and documentation.

Cost index

<table>
<thead>
<tr>
<th>Safety Level</th>
<th>Low</th>
<th>Medium</th>
<th>High</th>
<th>Very high</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>0%</td>
<td>20%</td>
<td>40%</td>
<td>80%</td>
</tr>
<tr>
<td>High maturity</td>
<td>100%</td>
<td>80%</td>
<td>60%</td>
<td>40%</td>
</tr>
</tbody>
</table>

Four Bold Requirements Engineering Themes

Value Innovation

Product management

Collaboration

Quality requirements

Collaboration – Challenge

80% of distributed development cause problems
20% of sourcing contracts are stopped during first year
50% of sourcing contracts do not achieve goals and are cancelled

Causes:
- Unsuitable sourcing models
- Unqualified suppliers
- Communication problems
- Incomplete specifications
- Insufficient process and collaboration capabilities

Collaboration – Solution

Vector recommendations:
- Implement requirements collaboration strategy for company and suppliers
- Use PLM with open interfaces (e.g., ReqIF) to share and cooperate during requirements engineering
- Use consistent templates on the level of single requirements and entire specifications
- Check requirements quality automatically

Leader

Innovate and protect
Lead and collaborate

Follower

Prepare and adapt
Pick and play

Internal

External
Collaboration – Benefits

Collaboration EBIT potentials (conservative)

<table>
<thead>
<tr>
<th>Current EBIT</th>
<th>EBIT Potential</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Shortened cycle time**
  - Time to release
  - Rework
  - Schedule adherence
  - Reuse

- **Reduced cost**
  - Less corrections,
  - Changes,
  - Variants

- **Improved quality**
  - Release criteria
  - Less defect numbers,
  - Corrections in field

Vector project experiences:
- Improved collaboration with PLM (process, tools, teams) allows for a 10% EBIT improvement, mostly by cycle time and cost reduction.

Collaboration – Benefits

Vector project experiences:
- Improved collaboration with PLM (process, tools, teams) allows for a 10% EBIT improvement, mostly by cycle time and cost reduction.

Four Bold Requirements Engineering Themes

- **Value innovation**
- **Product management**
- **Collaboration**
- **Quality requirements**

Four Bold Requirements Engineering Themes

Value Innovation – Challenge

- **Value**
  - Realistic
  - Good enough
  - Target
- **Effort**
  - Overengineering
  - Insufficient quality
  - Technical debt

Value Innovation – Challenge

- **Effect**
  - Customer satisfaction
  - Excitement factors
  - Performance factors
- **Completion of features**
- **Base factors**

Value Innovation – Solution

-vector recommendations:
- Focus on value. First understand the customers’ business case.
- Apply systematic requirements engineering for value innovation.
- Make requirements measurable and relate to the business case.
- Empower the team. Make people responsible for the outcome.
- Eliminate waste. Consider pay-off for each activity.
Value Innovation – Benefits

Vector project experiences:
- Introduction of lean development supported by feature-driven development and frequent reviews of portfolio, product, requirements.
- Result after 2 years: Reduction of cycle time by 10-35%, reduction of development cost by 10-15%.

Productivity vs. Requirement Stability

Optimize value creation with requirements engineering

- How to reduce effort and cost during the initial creation period?
- How to accelerate the move from creation to capturing?
- How to maximize the value during the life-cycle?

Agenda

- Snapshot on Requirements Engineering
- Industry Challenges
- Applying Requirements Engineering
- Delivering Results

Four Bold Requirements Engineering Themes – Benefits

- Value innovation
  - Cycle time: ↓ 10-35%
  - Development cost: ↓ 10-15%

- Product management
  - Market entry: ↓ 20-30%
  - Defects, delays: ↓ 50-80%
  - Customer satisfaction: ↑ 20%

- Collaboration
  - EBIT: ↑ 10%

- Quality requirements
  - Life-cycle cost: ↓ 10-20%
**Requirements Engineering: Trends**

**Society and Business trends**
- Global competition
- Cost pressure
- Demand for value
- Ever-changing expectations
- Service
- Individualism
- Green
- Security

**Impacts on Requirements Engineering**

**Strategy:**
- Value. Make customers part of value creation
- Segmentation: Dynamically segment to the single-buyer
- Solutions: Manage solutions with products and services

**Operations:**
- Cost. Use lean and agile principles
- Quality. Establish disciplined processes
- People. Effectively manage competences and knowledge

---

**Improve Your Requirements Engineering to Deliver Value**

**The real challenge for our clients is the culture change**
- Industry-specific cultures hamper innovation and efficiency
- Value-creation and cost of complexity are not understood (e.g., variants, processes, suppliers)

**Implement focused change**
- Motivation: How to make people burn to do better?
- Process: How to keep projects in time and budget?
- Quality: How to deliver worldwide the right quality?
- Productivity: How to create more value per unit of labor?

**Optimize your requirements engineering with concrete goals.**

---

**Change Your Focus from Inward to Outbound**

- Sell your product / idea with passion
- Create compelling vision
- Perceive the product and its value from the customer’s perspective
- Get stakeholders early on board, use the “core team” approach
- Speak business language to communicate value
- Compare with competitors’ products
- Learn from failures

**First focus on WHAT sells. Then engineer HOW to innovate and implement.**

---

**More Background …**

**Requirements Engineering**

By Christof Ebert
IEEE, Build Your Career TechSet
http://www2.computer.org/portal/web/buildyourcareer/techsets

Three online overviews covering the topics of this tutorial:
- Requirements Engineering Primer
- Practical Requirements Engineering
- Requirements Engineering: Advanced Topics
More Theory …

**Requirements Engineering: Processes and Techniques**

By Ian Sommerville, Gerald Kotonya
Hardcover (September 1998)
John Wiley & Son Ltd;
ISBN: 0471972088

Broad compendium of methods and theory of requirements management. Various modeling techniques (particularly those that came before UML) are presented and explained in a practical way.

---

Value Orientation

**Inspired: How To Create Products Customers Love**

Marty Cagan
SVPG Press, 2008

Why do some products make the leap to greatness while others do not? Creating inspiring products begins with discovering a product that is valuable, usable, and feasible. If you can not do this, then it s not worth building anything.

The book is written with a product management background, but very much emphasizing what are success factors, and what are real value points – exactly the theme of lean development and lean management.

---

R&D Value Creation

**Managing Research, Development and Innovation: Managing the Unmanageable**

Ravi Jain, Harry C. Triandis, Cynthia W. Weick
Wiley, 3 edition, 2010
ISBN: 978-0470404126

A very well done and thought provoking work to allow the reader to gain insight on the challenges and opportunities of managing research and development in a fast-paced technology world. Lots of insight to world class examples.

---

New Product Introduction

**Winning at New Products**

Robert G. Cooper
ISBN: 978-0465025787

Cooper offers the standard introduction to product development. Based on a broad data repository that Cooper and colleagues has collected from survey and consulting. Though the data is sometimes dated, he provides useful success factors for value generation, introducing new products and for managing innovation.
"This book stands out as the best source of information on distributed software development. Seldom do we see a book with the concepts completely backed by industry experiences and views. Software developers and managers benefit from the broad case studies."

S M Balasubramaniyan, Vice President, Wipro Technologies