



Fraunhofer Institut
Experimentelles
Software Engineering

Evolution of the ReqMan Framework

Authors:

Sebastian Adam
Tom Koenig
Thomas Olsson

IESE Report No. 057.06/D
Version 1.0
April 10, 2006

A publication of Fraunhofer IESE

Fraunhofer IESE is an institute of the Fraunhofer-Gesellschaft. The institute transfers innovative software development techniques, methods, and tools into industrial practice. It helps companies to build up demand-oriented software competencies and to achieve a competitive position on the market.

Fraunhofer IESE is directed by
Prof. Dr. Dieter Rombach (Executive Director)
Prof. Dr. Peter Liggesmeyer
Fraunhofer-Platz 1
67663 Kaiserslautern
Germany

Abstract

The ReqMan Framework is considered a theoretical, coarse structuring of concepts related to Requirements Engineering. For this purpose, the concepts of requirements phases, practices, and techniques were introduced in a first iteration, and defined based on a study of the literature. The first version of the Framework was evaluated in different application contexts with a focus on those practices that are considered the central instrument, with the intention of identifying potential optimization regarding its use. It became clear during the course of the evaluation that on the one hand, there are large differences between the practices developed with regard to their granularity, and, on the other hand, that these practices also permit ambiguous interpretations. In particular, across SMEs, the number of practices (47) was not manageable. Based on the results of the evaluation steps, a second evolution phase of the Framework (Framework 2.0) was developed and implemented. This report describes the results of the evaluation as well as the second evolution phase of the ReqMan Framework.

Keywords: ReqMan, framework, requirements, requirements engineering, requirements engineering process, evaluation

Table of Contents

1	Initial Situation	1
1.1	The Existing Framework	1
1.2	Criticism	2
2	Main Features of the New Framework	4
2.1	New Schema	4
2.2	New Set of Practices	5
3	Evaluation	8
3.1	Goal and Set-Up of the Evaluation	8
3.2	Quantitative Assessment	8
3.3	Qualitative Assessment	11
4	The ReqMan Framework 2.0	16
5	Summary and Outlook	18
6	Questionnaire	19

1 Initial Situation

1.1 The Existing Framework

In the first version of the ReqMan Framework, 47 practices that are important from the point of view of Requirements Engineering were categorized into the five process phases of RE typically found in the literature (elicitation, analysis, specification, verification and validation, management). In addition, they were classified into basic, extended, and context practices according to their use for the organization employing them.

15 of these practices were classified as basic practices, which were considered to be those principles and activities that are very important in any context and thus should be used.

Another 18 practices were classified into the category of extended practices, which were considered to be advanced activities that also offer great benefit, but cannot be used sensibly without the underlying basic practices.

The remaining 14 practices were classified as context practices, which cannot be evaluated regarding their importance outside the company or project context, and which also should only become part of a company's RE processes in a context-dependant manner.

Figure 1 provides an overview of the ReqMan Framework in its first version.

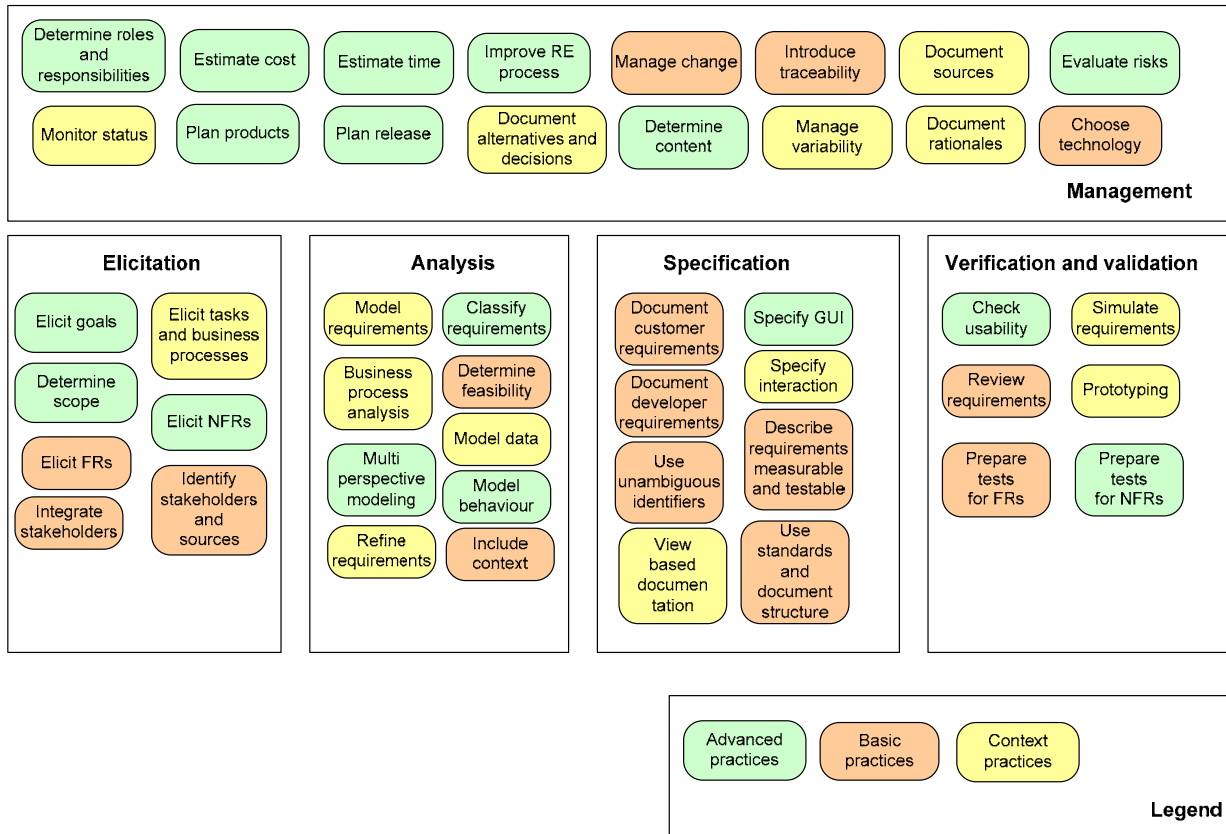


Figure 1 The existing Framework

1.2 Criticism

Both through the use of the Framework in case studies and through analytical observation, the following critical issues could be detected.

- Number of practices too high for SMEs

Many small or medium-sized software companies often only fulfill very few of the practices with regard to improving their processes with the help of the Framework (< 10). The idea of "What-one-should-actually-do" and what many smaller companies actually do may, in a sense, make them insecure in their Requirements Engineering activities and might even have a demotivating impact. Some practices are closely related to each other and are normally covered by common techniques. So far, this aspect was taken into consideration during the selection of the practices and techniques. By combining overlapping practices, it would be possible to reduce the amount of practices without having to

give up the principal benefit of the practice (the selection of concrete techniques).

- Domain-related weighting of context practices not considered

The procedure being followed during the existing use of the ReqMan Framework in the context of process improvements and assessments always consisted of first focusing on the basic practices, then on the extended practices, and finally on the context practices. This procedure implied the view that the context practices have the least importance in the area of Requirements Engineering. However, this is usually not true. For example, the practice "Elicit tasks and business processes" may have the importance of a basic practice in companies that develop information systems. This domain knowledge has only been considered implicitly so far.

- Basic practices show different levels of granularity

Many practices, especially basic practices, show different levels of granularity, which makes it more difficult to check the current state or to introduce suitable practices. On the one hand, there are simple principles that can be fulfilled without explicit techniques (e.g., "Use unambiguous identifiers"); on the other hand, there are complex practices that require extensive methods, including specific tools and process steps (e.g., "Manage changes").

- Phase assignment not always clear

The assignment of practices to the individual phases or their classification into the various types (basic, extended, context) is not always clear or logical. Thus, for example, practices dealing with the specification of certain aspects can be found under "Management", and several practices that are the prerequisite for other practices are classified as context practices, while the dependant practices are classified as extended practices.

2 Main Features of the New Framework

The basic idea for the new Framework was, on the one hand, to re-design the classification schema in such a way that better consideration can be given to the different weighting of (context) practices and the sequence of their introduction.

In addition to that, the number of practices was reduced and compressed and inserted into the new classification schema.

2.1 New Schema

As in the first version of the Framework, the categorization of the practices according to phases (elicitation, analysis, specification, verification & validation, management) was kept.

Different from before, however, the various kinds of practices (basic practice, extended practice, context practice) were grouped according to their importance by layering them in a new manner, and thus an implicit sequence of introduction was also defined (see Figure 2).

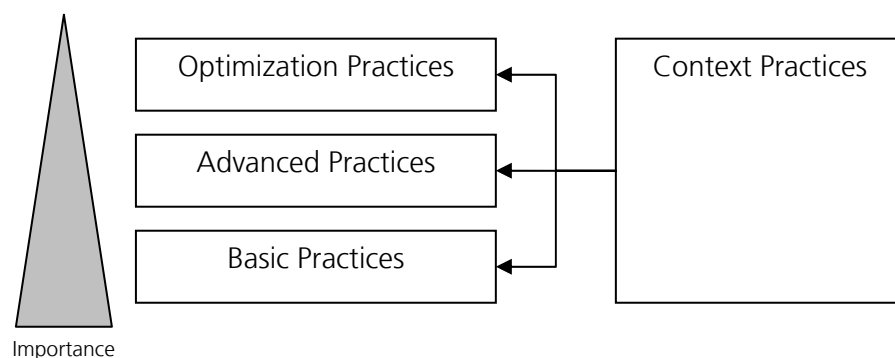


Figure 2

Basic schema of the new Framework

While in the first Framework, basic practices were considered to be those practices with a large degree of benefit, the new schema considers basic practices to be those that are absolutely required for being able to perform any initial Requirements Engineering at all (must-have-practices).

With the help of the advanced practices, this initial Requirements Engineering process can then be adapted to a specific company, and may be complemented by suitable quality and project management practices (should-have-practices).

Finally, optimization practices, meaning those that are not necessary per se, may be used to provide an additional, optimizing benefit to the using organization (nice-to-have-practices).

Different from the existing Framework, the context practices are not also classified into this hierarchy, but are considered orthogonally to this hierarchical layering. Depending on the particular company or project context, however, they are mapped to the weighting levels mentioned above. A company that develops software for the support of organizational processes, for example, might therefore consider the context practice "Elicit business processes" as a basic practice.

2.2 New Set of Practices

In order to limit the existing amount of practices, individual practices were completely removed from the Framework on purpose or were joined with other practices.

Joining was motivated by large overlaps or coverage through common techniques. Elimination, on the other hand, was motivated by a lack of practical importance or explicit implementability via specific techniques.

The resulting draft for the new version of the Framework is shown in

Figure 3. As can be readily seen, the number of practices was reduced especially in the area of "Management", but also in "Analysis" and "Elicitation".

The following statements provide a brief overview of new or changed practices.

- "Identify stakeholders and sources" integrates "Integrate stakeholders"
- "Model formally" is a new practice and contains the mathematical, formal analysis of requirements
- "Create GUI model" is a practice for deriving a usable user interface
- "Create domain model" is a new practice for analyzing the software environment, such as in the case of business processes

- “Create interaction model” is a new practice for analyzing the interaction of a system with its users or with other systems
- “Analyze requirements impact” is a new practice for checking on the impact on other products or on the project
- “Introduce traceability” is a new practice for tracking requirements implementations in later products
- “Document rationale” now also includes “Document sources”
- “Document customer requirements”, respectively “Document developer requirements” now also includes “Use unambiguous identifiers”
- “Formally check requirements” is a new practice for mathematical reasoning and also includes “Simulate requirements”
- “Prepare tests on requirements” now integrates “Prepare tests on functional requirements” and “Prepare tests on non-functional requirements”
- “Support requirements changes” is a new practice for fundamentally securing a change process
- “Estimate costs and time” now integrates “Estimate costs” and “Estimate time”
- “Plan product” now also includes “Plan release”
- “Reuse requirements” is a new practice and includes activities for reuse already on the level of requirements
- “Prioritize and negotiate requirements” is a new practice and expands the existing practice “Negotiate requirements” with the aspect of prioritization

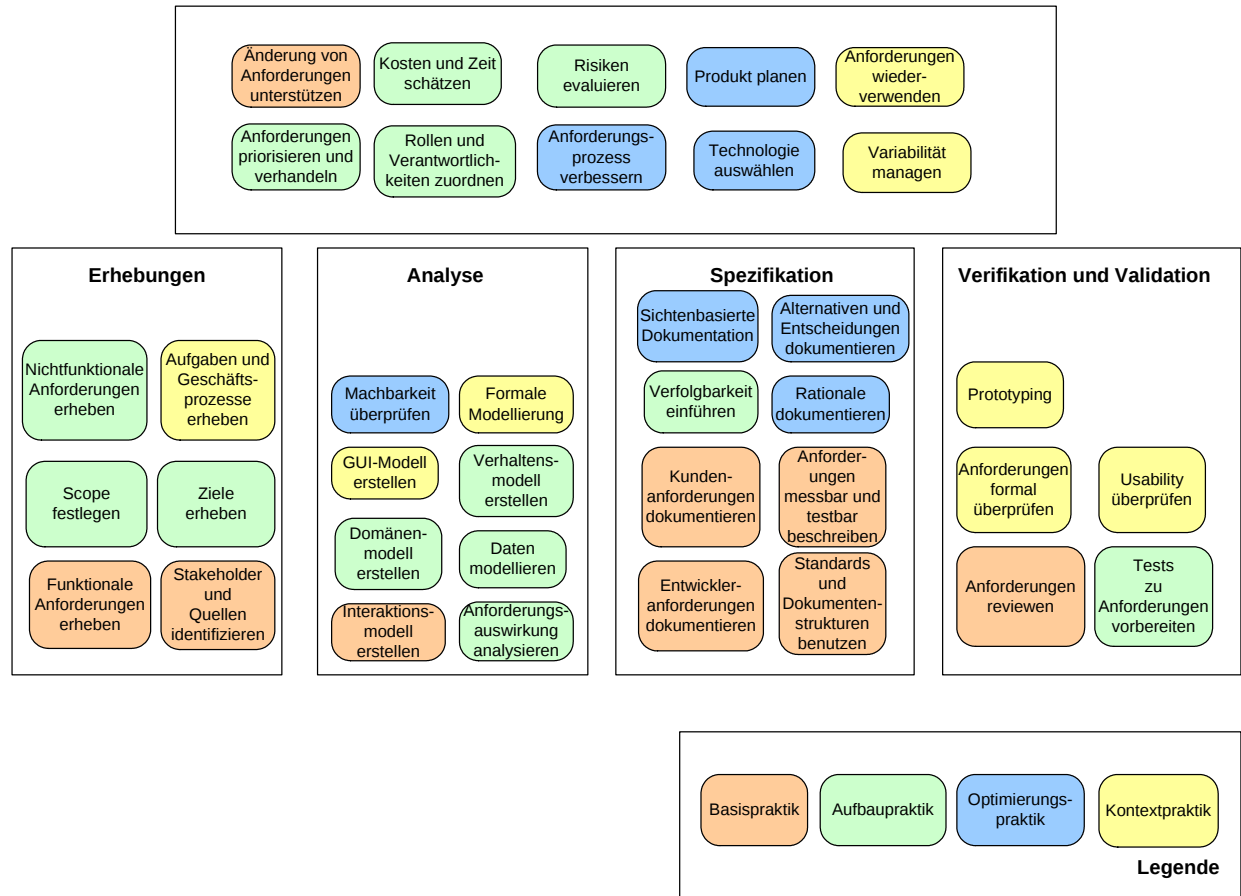


Figure 3

The new Framework before evaluation (only available in German)

3 Evaluation

3.1 Goal and Set-Up of the Evaluation

The goal of the evaluation was to achieve a quantitative confirmation of the classification of individual practices into the existing phases and levels of importance on the one hand, and to receive constructive feedback on individual practices as well as on the overall classification on the other hand.

All members of the IESE department RUE as well as those employees of the consortium partners methodpark, insiders technologies, and Fraunhofer IAO who were involved in the ReqMan project were included as participants in the study. Altogether, 14 data points could thus be obtained.

The table shown in chapter 6 was used as study material. The participants in the study were expected to classify each practice into exactly one level and one phase, and, if applicable, state their criticism of the practice (e.g., "cannot be classified clearly", "ambiguous", etc.). At the end of the table, there was extra space for free comments on the overall approach.

3.2 Quantitative Assessment

The results of the quantitative assessment are summarized in Table 1.

	Phases					Importance			
	Elicitation	Analysis	Specification	V&V	Management	Basic Practice	Advanced Practice	Optimization Practice	Context Practice
Support requirements changes					14	10	2	1	1
Formally check requirements				14				3	11
Describe requirements in a measurable and testable way		4	10			7	6	1	
Prioritize and negotiate requirements	3	2		2	7	8	6		
Review requirements				14		12	2		
Reuse requirements	2		1		10			5	9
Analyze requirements		10		1	2	1	11		1
Improve requirements process					13	1	3	9	
Elicit tasks and business processes	14					5	1	1	7

Model data		9	5			5	7	1	1
Create domain model	3	8	3			3	7		3
Document developer requirements			14			7	4	2	1
Model formally		6	8				2	1	11
Elicit functional requirements	14					14			
Create GUI model		6	8				2	1	11
Create interaction model		11	3			2	7	2	3
Estimate costs and time		1			13	7	7		
Document customer requirements	2	1	11			14			
Check feasibility		9		4	1	7	5	1	1
Elicit non-functional requirements	14					9	5		
Plan product	1				14	4	1	8	3
Prototyping	2		2	8			6	5	3
Document rationale	1		10		1	1	5	5	1
Evaluate risks		4			9	4	8	2	
Assign roles and responsibilities			1		13	4	8	1	
Determine scope	7	4	1		1	9	3		1
Perspective-based documentation			14			1	2	11	
Identify stakeholders and sources	14					11	3		
Use standards and document structures	1		13			9	5		
Select technology		3	1		9	3	4	5	
Prepare tests on requirements		1	2	11		2	8	4	
Check usability		1		13			4	1	9
Manage variability		3	1		10			3	11
Ensure traceability			7		7	1	8	5	
Create usage model	2	10	3			1	8	3	2
Elicit goals	14					8	6		

Table 1 Assessment of the quantitative study

While some assignments have a high degree of correlation (>70%), the classification is highly differentiated in the case of some practices. This refers mainly to the classification with regard to the importance.

Therefore, decisions by the ReqMan project team regarding the classification were necessary for determining the future Framework. The results of the evaluation thus primarily offer only an orientation for determining the new version of the Framework. The decisions that were actually made are visualized by cells with a grey background.

In the following, deviations between the results of the survey and the decisions made are explained.

- Model formally

“Model formally” was understood by the majority of those surveyed as being a specification practice. However, the decision was made to consider all modeling activities as analysis practices.
- Create GUI model

Although most of those surveyed also classified this practice as a specification practice, it was classified as an analysis practice for the same reason as in the case of “Model formally”.
- Estimate costs and time

“Estimate costs and time” was assessed by an equal number of people as a basic practice and as an advanced practice. However, since this practice does not illustrate any causative areas of Requirements Engineering, the practice was classified as an advanced practice.
- Ensure traceability

This practice was classified by 50% each of those surveyed as “Specification” and “Management”. Since, however, traceability is usually ensured by means of appropriate relation documentation, the practice was considered to be a specification practice.
- Describe requirements in a measurable and testable way

The practice sounds very simple, but requires a lot of practice and discipline. Furthermore, especially for small companies it is important to even describe requirements at all (ad hoc). Therefore, despite a small majority in favor of it being a basic practice, this practice has “only” been classified as an advanced practice.
- Prioritize and negotiate requirements

This practice was considered to be a basic practice by the majority of those surveyed. The reason for classifying it nonetheless as an advanced practice is that it does not address any of the more severe Requirements Engineering problems.
- Check feasibility

“Check feasibility” was also considered a basic practice by the majority. However, since it is a requirements problem only indirectly (rather being a project problem), and since it is also dependant on decent requirements management, it is classified as an advanced practice.

- Prototyping
 “Prototyping” was considered to be an advanced practice by the majority. However, since it does not belong to the classic verification and validation practices (in contrast to reviews and tests), it was classified as an optimization practice.
- Document rationale
 This practice was selected as an advanced practice and as an optimization practice by the same number of respondents. It was decided to classify it as an optimization practice.

3.3 Qualitative Assessment

The following issues were raised as qualitative results of the survey. The notes describe answers of the ReqMan project team to aspects that were voiced.

- “Document alternatives and decisions” and “Document rationale” should be combined.
 Note: This was done following the evaluation.
- “Support requirements changes” should be renamed to “Define change process”.
 Note: This was also done.
- “Elicit tasks and business processes” should be separated, since there are always tasks that are supported by a system, but there are not always business processes.
 Note: Here, there is a difference in the understanding and in the definition. Since there was only one comment on this, the decision was made to leave the definition unchanged.
- “Create domain model” is not clear
 Note: There is a difference in understanding on this issue as well. Since there was only one comment made on this, the decision was made to leave the definition unchanged.
- “Create behavior model” is not clear, since the term “behavior model” focuses on system behavior, whereas in this case, a usage model is actually meant.
 Note: The practice was renamed to “Create usage model”.

- The Framework is not complete. For example, there are practices to document an aspect, but there is no practice to elicit it.

Note: The Framework intends to be complete, and none of the remarks so far indicates that this is not the case. However, it is possible to see where this criticism comes from, since the different phases are not categorized in the same way.

- The phases are not defined clearly and disjunctively.

Note: This is indeed true, but it also reflects the reality. The problem is more of a theoretical nature, and not so relevant for practice. There is awareness of this issue, but no solution for it.

- Context practices must be categorized according to importance, so that one knows to which level of importance they are assigned in a given context.

Note: Internal classification according to importance is planned for the near future.

- The importance classification should be defined through the causal dependency of practices.

Note: Visualization and assessment of causal relations between the practices is certainly helpful and will be used for determining the logical introduction sequence of practices.

- A practice "Define RE process" is missing.

Note: This is covered by the practice "Improve requirements process".

- The focus of the practices should be more on elicitation and less on specification, since this is the lesser problem.

Note: it certainly is the case that different foci exist in different environments. The Framework is intended to cover as much context as possible and deal with the practice-relevant problems. It is probably impossible to find one classification that is acceptable for everyone. However, concrete suggestions are very welcome!

- User-oriented practices are missing.

Note: This is indeed the case and should be resolved by including appropriate practices.

- Importance should not be determined on the level of practices, but only on the level of techniques.

Note: There is importance on both levels. The Framework attempts to represent the more abstract and more generally valid conclusions. The hypothesis is that the importance of techniques can easily be assessed by the company itself, whereas this is not necessarily the case for the importance of the practices.

- The practices should be classified according to decision points (TORE) instead of phases.

Note: This is certainly another interesting classification and will be investigated (additionally) during the course of further research.

- The portal's target group should define the phases.

Note: Primarily, one should pay attention to the fact that the ReqMan Framework and the portal do not necessarily need to be the same. However, it is indeed a good idea to make the decision on the phases via a survey (e.g., selection of given definitions).

- Analysis and elicitation are given insufficient attention by the practices.

Note: see "The focus should be more on elicitation."

- Most of the practices cannot be sensibly assigned to exactly one phase.

Note: This problem is known. Classification of the practices into phases is therefore done according to the core concept, that is, according to the focus of a practice.

- By being assigned to a certain phase, a practice is being pushed into a certain direction regarding its importance.

Note: In order to keep the underlying concept simple, the decision was made that one practice only belongs to one phase. Of course, this simplification results in shortcomings in other areas. The criticism voiced is one example. However, at this time there is no better solution to this issue.

- The assessment of the importance of a practice can only be provided with regard to the respective phase.

Note: see previous criticism.

- The number of practices is still too large.

Note: Unfortunately, given the objectives of the Framework, it is not possible to further reduce the number. Many factors need to be taken into account and therefore, a compromise with respect to number and completeness is necessary.

- The target group, i.e., the visitors of the portal, should be included in the evaluation.

Note: For the third and last version of the Framework, there will be a major evaluation and involvement of external experts.

- A practice "Understand project character" would make sense.

Note: In the Framework, the project character is understood as a context condition and not as a practice.

- The description of the practices is very abstract and does not provide much information.

Note: The description of the practices will be completed and improved with regard to their understandability.

- The Framework contains practices that actually do not originate in the area of Requirements Engineering.

Note: The Framework is intended to offer a pragmatic building kit for smaller and medium-sized enterprises and thus also contains important project and quality management practices that are significant with regard to Requirements Management.

- The phase "Analysis" is not clear, since analysis is no end in itself, but rather always serves other phases.

Note: There are many definitions of requirements management and the phases that belong to it. Here, the widely used 5-phase model, which also contains analysis, was selected.

- The importance of practices must always be assessed only in the given usage context.

Note: The hypothesis is that most of the practices can also be assessed with regard to their importance without context information. In order to prove this, the Framework is being evaluated in several case studies during the ReqMan project. This issue thus still remains open.

- In the Framework, “Management” is not used in the sense of “Management” as understood by the Requirements Engineering community.

Note: In the ReqMan Framework, “Management” is a collection of practices that deal with the administration as well as with the systematic control of the requirements process and its products. Due to the pragmatic character, this phase also includes practices that are not typical of RE.

- Instead of assigning one practice to exactly one phase and exactly one level of importance, one should describe a classification that visualizes the use of individual practices in different phases in a way similar to the diagrams used in RUP.

Note: Visualization does indeed offer a sensible overview of the usage of practices in individual phases. However, in order to keep the underlying concepts simple, it was decided that one practice should only belong to one phase.

- The names of many practices are not self-explanatory and lead to confusion.

Note: Making decisions about names, selections, and descriptions is a very difficult and far-reaching job. However, it cannot be expected that all practices can be completely understood without a description. Nonetheless, concrete input is welcome.

4 The ReqMan Framework 2.0

Figure 4 provides an overview of the new version of the Framework after its evaluation.

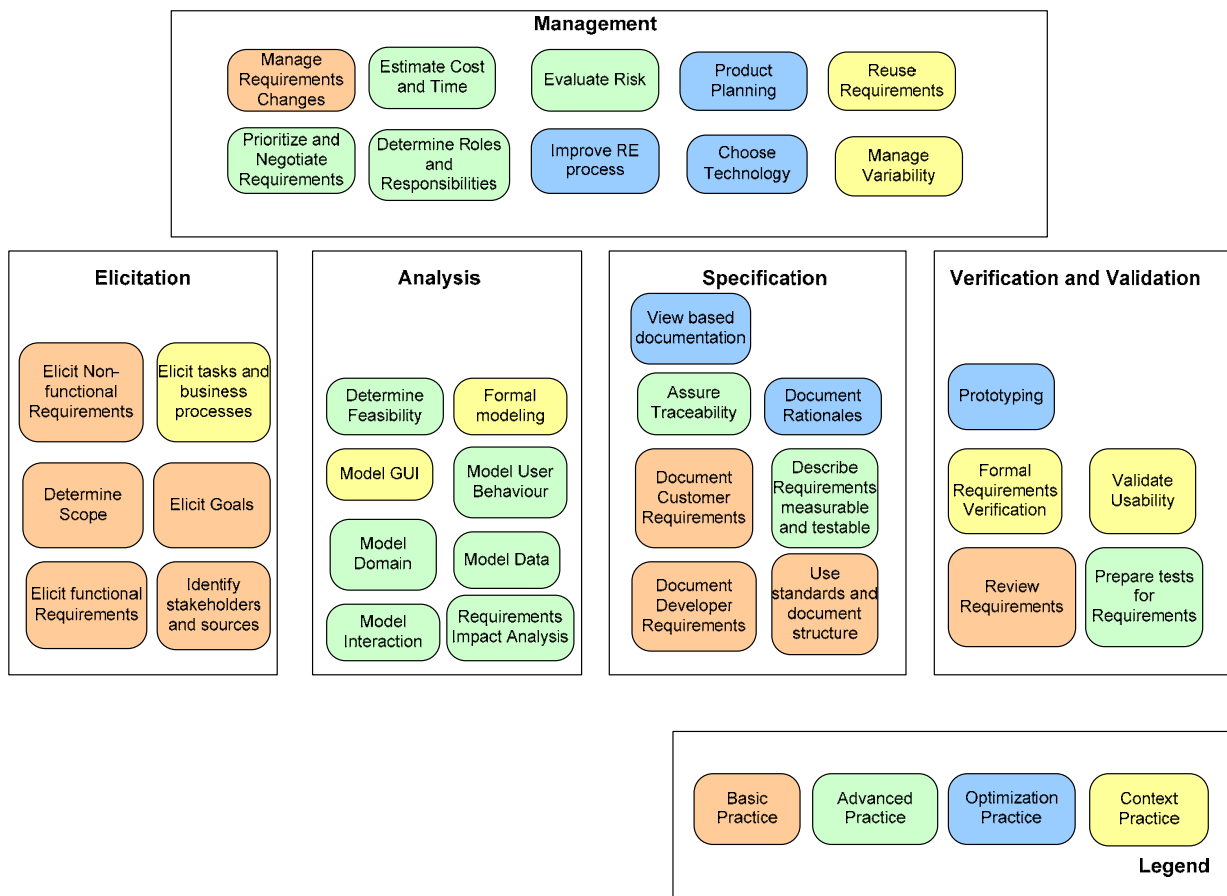


Figure 4 The new Framework after its evaluation (official version 2.0)

The following differences can be reported as changes with regard to the new Framework as originally planned:

- Elicitation
With the exception of "Elicit tasks and business processes", all other practices have now been classified as basic practices.

- Analysis

“Create behavior model” was renamed to “Create usage model”. “Check feasibility” and “Create interaction model” were classified as an advanced practice.

- Specification

“Document alternatives and decisions” was classified under “Document rationale”. “Describe requirements in a measurable and testable way” was classified as an advanced practice.

- Verification & Validation

“Prototyping” is now an optimization practice.

- Management

“Support requirements changes” was renamed to “Define change process”.

5 Summary and Outlook

This report describes the evaluation of the first phase of the ReqMan Framework and presents the second evolution phase of the ReqMan Framework, Framework 2.0. Initial assessments regarding the Framework already indicate the optimizing character of the new evolution phase. However, so far, the evolution was performed with a limited number of participants, in particular, with representatives from a scientific environment. In another iteration phase, an evaluation shall be performed that involves an expanded circle of participants, with particular emphasis on the industrial environment, ending in the 3rd evolution phase of the ReqMan Framework. Various evolution activities are planned, which are briefly listed and explained below:

- Phase definition by visitors to the portal: Visitors of the portal re-wissen.de shall make a selection from a given number of definitions and thus reach a consensus.
- Classification of the ReqMan practices in the context of another survey (see 3.1) with a wider circle of test subjects (possibly via the re-wissen.de portal).
- Refinement of the mapping of the context practices: Identification and definition of relevant context factors that allow an assessment of the importance of a practice in a context (thus its mapping).
- Focus on visualization: Visualization of the ReqMan Framework, especially for clarifying the correlations between practices, between techniques, as well as between practices and techniques.
- Extension, tailoring of the questionnaire/checklist: The existing RE Checkup questionnaire, which was already adapted to version 1.0, shall be further tailored to the new evolution phases during the course of the work. In particular, it shall also be expanded and refined.

This list represents the central tasks to be performed during the course of the second iteration phase, which will lead to the third evolution phase of the portal.

6 Questionnaire

Evaluation of the ReqMan Classification

	Phases						Importance				
	Elicitation	Analysis	Specification	V&V	Management	Cannot classify	Basic Practice	Advanced Practice	Optimization Practice	Context Practice	Cannot classify
Please check only one phase field and one importance field per practice! Should the practice fall into several categories, please select the category where the main focus lies. If you still cannot classify a practice, select the respective "Cannot classify" field.											
Document alternatives and decisions											
Support requirements changes											
Refine requirements											
Formally check requirements											
Classify requirements											
Describe requirements in a measurable and testable way											
Prioritize and negotiate requirements											
Review requirements											
Reuse requirements											
Analyze requirements impact											
Improve requirements process											
Elicit tasks and business processes											
Model data											
Create domain model											
Document developer requirements											
Model formally											

Questionnaire

Elicit functional requirements														
Create GUI model														
Create interaction model														
Estimate costs and time														
Document customer requirements														
Check feasibility														
Elicit non-functional requirements														
Plan product														
Prototyping														
Document rationale														
Evaluate risks														
Assign roles and responsibilities														
Determine scope														
Perspective-based documentation														
Identify stakeholders and sources														
Use standards and document structures														
Select technology														
Prepare tests on requirements														
Check usability														
Manage variability														
Introduce traceability														
Create behavior model														
Elicit goals														

Document Information

Title: Evolution of the ReqMan Framework

Date: April 10, 2006

Report: IESE-057.06/E

Status: Final

Classification: Public

Copyright 2006, Fraunhofer IESE.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means including, without limitation, photocopying, recording, or otherwise, neither wholly nor in excerpts, without the prior written permission of the publisher. Written permission is not needed if this publication is copied or distributed for non-commercial purposes.