PATTERN: Requirements Pyramid

ANNE HOFFMANN, Siemens AG, Munich, Germany

Categories and Subject Descriptors: C.0 [Computer Systems Organization GENERAL] Modeling of computer architecture; D.2.1 [SOFTWARE ENGINEERING] Requirements/Specifications ; H.1 [MODELS AND PRINCIPLES]

General Terms: RE Pyramid, Requirements Engineering Pyramid

Additional Key Words and Phrases: Frontloading, Stakeholder Management

1. INTRODUCTION

When joining a project, you are often confronted by a lot of information.

In the discipline of Requirements Engineering there is a known solution to overcome the arising issue of needing to access this information as quick as possible. As we experienced the REQUIREMENTS PYRAMID to be useful, but often not known by other roles such as architects, we decided to put these practices into a pattern, making it available for everyone.

Roles in projects, that could benefit from knowing this pattern among others are: project managers, requirements engineers, architects, business analysts, product life cycle managers.

Please note, that in the discipline of requirements engineering, every single piece of information is called a requirement.

You might now ask yourself what the profit of knowing the concepts of the REQUIREMENTS PYRAMID is. Assuming you are an architect who just had joined a running project. Now, you are trying to figure out why the architecture looks like it does look like. You might ask questions such as: "What requirements made us coming up with this solution and not with any other?" - Here, the REQUIREMENTS PYRAMID offers support.

By looking at Figure 1 you find the requirement "DR.OS.C.1 The camera shall support continuous shooting.". To understand, why this requirement is to be developed, you simply look upwards the pyramid, meaning following the upwards arrow in the figure until you find the corresponding requirements on the upper level. Thus, you get to know, that DR.OS.C.1 is in place, because "The system shall use the OptSys components and patents." is stated in requirement "BR.BC.4".

As this example shows the REQUIREMENTS PYRAMID ia as a 'conceptual model' or 'mental model' that helps you to structure and understand information stemming from different stakeholders and often stated in different levels of detail.

1 Known synonyms are: REQUIREMENTS ENGINEERING PYRAMID, RE PYRAMID or simply the PYRAMID.
Once, you understood the underlying concept of the REQUIREMENTS PYRAMID you will find more requirements engineering related tasks (such as analysis, prioritization, and change request management) where the REQUIREMENTS PYRAMID also eases clarification and understanding.

This pattern however focuses on the REQUIREMENTS PYRAMID as a mental model to support orientation whenever you need to get a quick overview of many requirements (e.g. when entering a project).

2. REQUIREMENTS PYRAMID

2.1 Context

Successful projects have one thing in common: They truly understood their stakeholders needs. This means, they detected all relevant requirements from their stakeholders. Stakeholders don’t structure their requirements. They simply state in their own words, what their wishes are. Every stakeholder is different and so are the requirements. This means that the retrieved set of requirements are as heterogeneous and different as the interests of the stakeholders. And, of course, their conflicting needs shall be detected as early as possible.

The larger and the more complex the project is, the higher become the need of keeping the overview. The REQUIREMENTS PYRAMID helps you to derive such an overview. It enables you to structure the requirements such that you retrieve a clear understanding of the project deliverables you need to satisfy the stakeholders.

2.2 Problem

In your project you have many requirements (information) from different stakeholders to satisfy but you build one solution only.

How can this manifold, inhomogeneous and partially conflicting input be converted into a balanced and complete set of requirements that provides you an overview and an easy way of understanding of the product to be developed?

When trying to answer this, you will be facing the following forces:
—Conflicts resulting from different requirements versus identify duplicates
Comparison of the collected requirements is needed to identify similarities and detect conflicts between any two requirements, finding and resolving conflicting requirements.

—Detailed versus abstract formulated requirements (Level of detail)
The level of detail of the information given to you differs, as requirements are formulated on different levels of detail. This makes comparison between requirements difficult and hinders detection of conflicts and later on understanding the impact of any requirement with the least amount of effort. This makes it hard to compare requirements collected from different stakeholders in the first place. In addition, requirements have to be linked with each other in order to allow traceability of information.

—Internal organizational constraints versus freedom of development
Some internal organizational constraints that might have been given to you explicitly or implicitly by stakeholders such as the business department of your company have to be respected. This could include the limit on overall cost, the programming languages to be used, market issues or regulations.
These have to be included as requirements in the REQUIREMENTS PYRAMID as well, as they are often enforcing decisions in a certain direction.

—Given solutions versus formulated problems
You have to understand the scope of your project as its collected requirements tend to be fuzzy, unstructured, and hard to compare. Often, some of these requirements are already solutions, representing a problem, the stakeholder was not or only slightly aware of. For any requirement being a solution this has to be analyzed in order to ensure that the product satisfies the stakeholder in the right way later on.

2.3 Solution
Requirements depend on each other, and have to be detailed from a top down view, where on the higher level Ū where the requirements are rather abstract Ū there are just a few requirements, and on the lower level Ū where the requirements tend to be very specific Ū there are many. This gives the model the shape of a pyramid. Hence, this is why it is called REQUIREMENTS PYRAMID. In summary, the REQUIREMENTS PYRAMID as a solution for the described problems, it
—uses hierarchy to define levels,
—assigns requirements to levels, and
—links requirements to each other (as well on the same as between to neighboring levels) thus reflecting interrelations such as one being the refinement or solution of the other.

So, in the ideal world, your REQUIREMENTS PYRAMID would follow a hierarchy, starting with a rather abstract requirements such as the wish for the product formulated in the stakeholder request, followed by a so-called marketing description or customer requirement, then assigned to a system, down to design specific requirements (or even solutions).
If requirements have a relationship, they are linked with each other. Having linked requirements enables tracing from a top level of the REQUIREMENTS PYRAMID to the corresponding bottom element(s) of the REQUIREMENTS PYRAMID and vice versa. On intermediate level every requirement is connected upwards and downwards.

2.4 Recommendation
Ideally, you would apply the REQUIREMENTS PYRAMID in the early phases of development. Then, whenever questions such as “Why did we make this decision?”, “Why do we have to implement this requirement?” occur, the REQUIREMENTS PYRAMID could be used to answer them.
You could use the REQUIREMENTS PYRAMID as a communication tool to communicate your view on the requirements to you colleagues, or you could even go as far as using it to derive a common understanding.
However, it is just a model. In real life, there are some typical observations:

— The REQUIREMENTS PYRAMID is almost never having a single top requirement.

— The REQUIREMENTS PYRAMID will often not be complete, as the solution to be developed is not purely an innovation, but known components are reused.

— The REQUIREMENTS PYRAMID is often cut in between even if only for some components. These "end"s should then be marked with a corresponding remark. In-between cuts occur for example when you are working with suppliers, as the point where the supplier’s responsibility starts is often defined on different levels of detail.

2.5 Implementation

An outline of a REQUIREMENTS PYRAMID is shown in the Figure 2.

Parallel lines divide a REQUIREMENTS PYRAMID into different levels. As seen in Figure 1 a requirement is put into its level, then an arrow (a so-called trace) links requirements, representing the relationship (e.g. a refinement) these requirements have.

Often, you might want to start with a REQUIREMENTS PYRAMID used with the following levels (starting from top): Stakeholder Requests, Customer Requirements, System Requirements, and System and Subsystem Design.

Depending on the project and / or the context you might want to use other level.

To ease the structuring, it is helpful to align a key question that should be answered by requirements aligned to this level, e.g. in Figure 1 on the System Requirements Level the question "What are the customer expectations?" is answered. Often, a certain specification type is assigned to each level, e.g. "System and Subsystem" could be collect in the architecture specification.

The idea is, that any requirement is linked in-between a requirement on the top and a requirement on the bottom of the REQUIREMENTS PYRAMID. The REQUIREMENTS PYRAMID is then called complete. The lowest level of requirements is then the detailed description (e.g. the design decision). then linked to the top level of the design which is solution specific.

Due to the high number of requirements and their manifold relationships it is good practice to

— Represent defined levels of the REQUIREMENTS PYRAMID within Requirements Management tools.

— Use attributes and views in Requirements Management tools to visualize requirements stakeholder specific, and

2.6 Consequences

Using the REQUIREMENTS PYRAMID in your projects results into the effect benefits: Through the linkage of requirements, it supports you

— to detect interrelations as requirements are ordered into levels,
— make decisions,
— prioritize requirements through the source of a requirement that it is linked with, and
— detect (potential) conflicts as requirements on the same or similar level are grouped together.

Through the links you retrieve the so-called traceability. If your links are complete, it is now very easy for you to identify

— requirements on a lower level without any predecessor on top level (the so-called "Gold plating"),
— Requirements on the upper level, not applied to the lower levels, thus not covered by the product (the so-called "Forgotten feature"), and
— any two requirements on upper levels whose combination of implementation is less than the sum of each of it (the so-called "Feature interaction").

In order to use the REQUIREMENTS PYRAMID efficiently, you should be aware of the following liabilities:
— That the Requirements Pyramid is used as "matter of course": One stops thinking when developing.
— Of over-engineering: It is difficult to find the just enough. How complete shall the Requirements Pyramid be? How many links do you have to detect, until you gained sufficient information? What are you considering "complete" in your project?
— Large organizations tend to draw department lines between the levels, thus discussions on requirement relationships become collaboration issues.
— Deflect the discussion from content-related questions to questions how to arrange of requirements into the different levels.

**Remark** In the Requirements Pyramid any requirement is as important as the other. The levels of the Requirements Pyramid do not provide any information about importance or priority.

2.7 Variant
— There is a $good practice$ in the Requirements Elicitation phase: If you use focus groups to discover or derive requirements for a new product or product version, the facilitator could use the guiding questions of the levels of the Requirements Pyramid in order to convey the derivation of the requirements.

2.8 Known Uses
The Requirements Pyramid is explicitly described in the requirements engineering literature, for example in [Berenbach and Paulish 2009, p. 42] Also, it is in use by trainers for Requirements Engineering. For instance, within Siemens Corporate Technology, we use the Requirements Pyramid in several trainings to ease understanding, and so do other companies focusing on Requirements Engineering consulting activities. For example see training material from [Ebert 2005, slide 118] and [Sie , p. 11].

3. FUTURE WORK (PATTERN)
In this pattern we introduced the Requirements Pyramid as model to structure information in a balanced manner to better under faster retrieve the overall picture when entering a project.

The approach of the Requirements Pyramid can even further been used as the basis for analyzing the requirements, deriving their dependencies and detecting deficiencies in the modeling of a product or in relation to completeness. This pattern can therefore be seen as the base for further patterns both in the area of Requirements Pyramid usage and of other requirements related patterns. There are more good practices related to this concept such as establish context-sensitive relationships between any two requirements and thus enriching the information kept.

So, in the future we think about formulation patterns such as a stakeholder-specific pyramid.

ACKNOWLEDGMENT
I would like to thank my shepherds Alexander Ernst and Philipp Bachmann.

Also, I would like to thank all participants from my workshop group at EuroPLoP 2010. In addition, would like to thank my colleague Dietmar Schütz for introducing patterns to me. Special thanks to Claudius Link for his support to strengthen the pyramid's forces.

REFERENCES
The idea of the Requirements Pyramid is widely used, following references are a randomly chosen extract of books and training material. Siemens internal training material slide set: Requirements engineering courses 2009. (Senior) Software Architecture Curriculum 2009.


