RePa Requirements Pattern Template

Lawrence Chung\textsuperscript{1}, Barbara Paech\textsuperscript{2}, Liping Zhao\textsuperscript{3}, Lin Liu\textsuperscript{4}, and Sam Supakkul\textsuperscript{5}

\textsuperscript{1}The University of Texas at Dallas, USA, chung@utdallas.edu
\textsuperscript{2}University of Heidelberg, Germany, paech@informatik.uni-heidelberg.de
\textsuperscript{3}University of Manchester, UK, lzhao@cs.man.ac.uk
\textsuperscript{4}Tsinghua University, China, linliu@tsinghua.edu.cn
\textsuperscript{5}Sabre Inc., USA, sam.supakkul@sabre.com

Submissions of specific requirements patterns to the International Workshop on Requirements Patterns (RePa) are required to be structured using this template. The use of a common template would help ensure the consistency among submissions, as well as facilitating the cataloging of patterns in a pattern repository.

This template has been derived and adapted from the literature for requirements patterns. It consists of three types of sections, two of which are described in this document, including required and optional sections, where required sections are denoted by an asterisk in front of the section names. Additional custom sections that the authors feel useful for the submitted patterns may be added. Examples of custom attributes may include consideration for development and consideration for testing [1].

The template we provide here can, and should, be tailored as needed. If the template is customized, e.g., not using the required (sub-)sections, or adding new (sub-)sections, your paper should discuss the rationale so as to advance our understanding of the use of pattern template for requirements patterns and consequently to help further refine the template.

Please note that technical report submissions are not required to use this template.

\*Name

The Name [2, 3, 5, 6, 9, 10, 14] provides a short descriptive handle for the pattern. A good name is essential as it will become part of the requirements engineering vocabulary [2, 3]. Examples of pattern names are “Detail the Specification by Writing Test Cases [4]” for a requirements process pattern, and “Watchdog” for a requirements product pattern [5].”

Also Known As

The Also Known As lists other names that may be used to identify the same captured knowledge [1, 3, 10].

Authors

The Authors [1, 6] identifies person(s) or organization(s) that documented this pattern, which can help provide credibility for the pattern [11].

\*Context

The Context [1-6, 9, 10, 14] describes the condition(s) under which the pattern is valid. It is also known as applicability [1, 3, 5].

- \*RE Activity: “Elicitation”, “Specification”, “Validation”, or “All”

RE Activity refers to requirements engineering activities where Elicitation refers to requirements elicitation activities such as domain analysis, negotiation and agreement [7] to determine what is needed, why needed, and whom and how to talk to acquire knowledge about the stakeholders’ needs. Specification refers requirements activities that document a (formal) requirements specification, artifact, product, or model to make the requirements explicit, complete, consistent, understandable, and precise, as well as requirements evolution that accommodate corrections, environmental changes, or new objectives [7]. Finally, Validation refers to activities that assure that the requirements specification satisfies the stakeholders’ needs.
• **Pattern Type:** “Process” or “Product”  
The Pattern Type identifies whether this pattern captures a requirements engineering process or product for the stated RE Activity. For instance, an Elicitation Process pattern may describe steps for conducting requirements gathering workshop, while an Elicitation Product pattern may capture a reusable questionnaire template. Similarly, a Validation Process pattern may describe steps for conducting requirements validation/review, while a Validation Product pattern may describe a prototype template or a traceability template.

• **Business Domains**  
The Business Domains describe the applicable real-world business or application domains, such as Financial Services or Travel and Hospitality. It is also known as owning domain [1]. Leave the section unspecified for domain independent patterns.

• **Organization Environmental Factors**  
The Organization Environmental Factors refer to internal and external factors that surround or influence a project, including but not limited to organizational structure, culture, human resources policies. It is also known as enterprise environmental factors [15] and application areas [4]. Examples are “small businesses with fewer than 50 employees” or “projects with large fluctuations in the team over time [4].”

• **Stakeholders**  
Stakeholders refer to the persons, roles, or organizations that are relevant to this pattern in some fashions. Examples of Stakeholders include but not limited to customers, users, sponsors, standard bodies, company developing the system in question, requirements engineers, and development, quality assurance, deployment, support and maintenance teams.

**Problem**  
The Problem [2-5, 9, 10] describes an undesirable situation faced by the stakeholders identified in the Stakeholders section. For example, customer may be faced with a problem of being unsure of the exact product features needed, or requirements engineer may be faced with a problem of not knowing how to conduct a requirements elicitation workshop. The Problem is also known as intent [3] and objective [4].

**Forces**  
The Forces [2, 3, 5, 9, 10] describe synergistic but oftentimes conflicting concerns that need to be balanced [2] by the requirements process or product. These potentially conflicting concerns are also known as (correlated) goals [8], side-effects [6], tradeoff [9], constraints [5], and rationale [9, 10, 18].

Examples of (conflicting) forces for a requirements process pattern are “In rooms, people are caught in a conflict: they are drawn to the chairs to sit down and relax but also to the windows where the light is [2]” and “The project has to step back to improve the specification” and “The project has to move forward to meet the milestones [4].”

It is observed that sometimes practitioners misuse Forces section to document organization environmental factors, such as “The team consists of experts who can lay the foundation and draw a road map to success”. It is therefore important to distinguish forces from context information.

The pattern should also document in this section the effects on these forces by the requirement process or product. The effects on forces are also known as resulting context [10], force resolution [9], consequences [3, 5], or goal achievement [8].

**Solution**  
The Solution [2-5, 9, 10] presents the requirements process or product that fulfills the stated intent and balances the stated forces.

If possible, pictures or diagrams representing the essence of the captured knowledge should be included to make the pattern easy to understand and use [2]. For example, a process diagram may be given for a
process pattern, while a requirements model may be given for a product pattern, to accompany the textual description.

**Application and Example**
The Application and Example [1, 2, 9, 10, 14] discuss how this pattern may be applied during requirements engineering. For example, a requirements product pattern may capture a requirements model, which may be applied during reuse by substituting the names of particular placeholder model elements with application specific model elements [3]. One or more examples may be given to illustrate the application [10].

**Known Uses**
The Known Uses [3, 4, 6, 14] describe the sources of the captured knowledge. For instance, Known Uses of a process pattern may describe systems or projects where the process has been applied successfully [2, 3]. Similar to the Authors information, works attributed to credible sources are viewed as more trustworthy [6].

**Cataloging**
- **Classification**
  For artifacts, such as patterns, to be effectively reused, it needs to be classified in multiple ways (faceted classification) [17]. The Classification [1, 3, 16, 18] should list applicable classifications with facet and classification name information. Examples of pattern classification include “Purpose: Creational [5],” “Purpose: Structural [5],” “Purpose: Behavioral [5],” “Security: Perimeter [16],” and “Security: Exterior [16],” where “Purpose” and “Security” represent facets while “Creational,” “Structural,” “Behavioral,” “Perimeter,” and “Exterior” are classification names respectively.

- **Related Patterns**
The Related Patterns [1-6] identify other patterns that are related to this pattern, each via a relationship. It is also known as pattern interaction [18]. For example, a pattern may be used by, similar to, combined with, refined by, or required by another pattern [12, 13]. A pattern may also be specialized by, aggregated by, or an occurrence of another pattern [6].

Examples for Related Patterns are “referred to by Controller Decompose pattern in its specification” (adapted from [5]), “must be used by X” (adapted from [12, 13]) and “is helped by Configured pattern [19].”

If multiple related patterns are presented in a paper, it is desirable that a diagram depicting the patterns and their relationships be also given outside the patterns to form a pattern language or system [2, 19].
Reference