

Business Patterns Catalogue and Selection Proposal for the Conceptual Model of a Software Product

Oscar Carlos Medina

(Universidad Tecnológica Nacional – Facultad Regional Córdoba, Argentina
 <https://orcid.org/0000-0003-3300-633X>, oscarcmedina@gmail.com)

Manuel Pérez Cota

(Universidad de Vigo, Pontevedra, España
 <https://orcid.org/0000-0003-0471-6981>, mpcota@uvigo.es)

Brenda Elizabeth Meloni

(Universidad Tecnológica Nacional – Facultad Regional Córdoba, Argentina
 <https://orcid.org/0000-0002-9721-4595>, bemeloni@gmail.com)

Marcelo Martín Marciszack

(Universidad Tecnológica Nacional – Facultad Regional Córdoba, Argentina
 <https://orcid.org/0000-0003-1398-6094>, marciszack@frc.utn.edu.ar)

Abstract: A pattern is a model that allows reusing a successful solution to the same problem in a different context. A pattern implementation could be the elaboration of an analysis model to incorporate good practices patterns Conceptual Modelling of Electronic Government systems. Defining a new pattern, and selecting a previously existing one from a limited set, called catalogue, are essential activities that every analysis model must solve when using patterns. The present work describes a proposal to manage a Business Patterns catalogue that can be applied to Conceptual Modelling of software products. Business Patterns allow to model and design business processes inside an organization, being it public or private. An application, called “PatCat” (Pattern Catalogue), was developed to test de proposal, using the Business Model of an Information System for a public education institution as a pilot. The introduction of patterns at the beginning of the Modelling Process allows to simplify and clarify the requirements elicitation, amongst other benefits. Thus, a specific management application for a pattern catalogue is useful to standardize and speed up this software design task.

Keywords: Business Patterns, Pattern Catalogue, Pattern selection, Conceptual Model, Software product, Education System, Electronic Government, eGov

Categories: D.2.13, D.2.2, H.1.1

DOI: 10.3897/jucs.65083

1 Introduction

There is a Software Engineering model that allows the generic description of a process in a simple, precise and standardized way, allowing to use previously gathered knowledge, and it’s called Pattern.

The concept of pattern was initially established by the architect Christopher Alexander, who exposes it in his books “A pattern language” [Alexander 1977] and

“The Timeless Way of Building” [Alexander 1979]. He defines that “*each pattern is a rule of three parts, that expresses a relation between a specific context, problem and solution*” [Alexander 1979].

Another foundational concept developed in these publications is the Pattern Catalogue, which is a set of patterns in the same domain as the problem. Even if Alexander’s intention was to create a structured method to extend architecture to non-specialized people, his accurate concepts allowed to be extrapolated to object-oriented software development domain fifteen years later.

Therefore, this pattern idea is used in software design by Gamma, Helm, Johnson and Vissides, who publishes the most important work about this subject: “Design Patterns: Elements of Reusable Object-Oriented Software” [Gamma et al. 1994], in accordance to Object-Oriented Programming paradigm. The second part, constituted by chapters three, four and five, introduces a catalogue of twenty-three patterns, which are all interconnected and classified in relation to their purpose in three categories, called “Creational”, “Structural” and “Behavioural”, and in two application domains: “Class Pattern” or “Object Pattern” [Gamma et al. 1994].

This book spreading originated patterns proposal for the different stages of software development, creating specific types for each one of them, as in the case of business processes patterns.

Patterns help leveraging the collective experience of specialized software, where engineers represent the existing and proven experience in systems development.

According to Eriksson y Penker [2], each pattern handles a determined recurring problem in the design or implementation of a software system. They state that Business Modelling Patterns allow obtaining the model resources, organizations, goals, processes and rules for the company. In their work, they present a catalogue of twenty-six business patterns for business modelling represented with UML, spreading their “*common business modelling experience and knowledge packaged in the form of reusable patterns (Resource and Rule Patterns, Goal Patterns, and Process Patterns)*” [Eriksson and Penker 2000].

A proposal to catalogue and choose Business Patterns for Conceptual Modelling of systems is developed in the present work, and it is experimentally implemented through “PatCat” (Pattern Catalogue). This is the name given to a web application that enables the management of a Business Pattern Catalogue applied to real E-Government cases in order to ease its construction by simplifying and clarifying the modelling stage.

2 Business Patterns Catalogue

2.1 Key Concepts

Electronic Government is “*the application of ICT (Information and Communication Technologies) to Government processes*” [Medina et al. 2018a]. ICT are considered “*as a dominant element to improve urban factors such as life quality, urban operation and services, and its economic, social and environmental supply*” [Estrada et al. 2018]. Also called E-Government or E-Gov (E prefix meaning “Electronic”), it contains information systems that support the Public Administration processes.

Good Governance Practices are related to these processes. Good Governance Practices are “*all initiatives and experiences that help improving the effectiveness of government actions and that positively affect the citizens’ quality of life, achieving a tangible impact on communities*” [Varela Rey 2016].

Good Governance Practices are used as reference in the construction of public software [CLAD 2015]. Thanks to the development of new technologies, “*the effectiveness and utility of electronic systems is being used to build a better e-service for citizens*” [Vo and Lai 2014].

These Good E-Gov Practices can be described with Business Patterns in the public software Conceptual Modelling stage.

The Conceptual Modelling of a system has the goal to identify and explain the important concepts of a problem domain, identifying the attributes and the existing relations between them. According to Sommerville, the Conceptual Model of the system is done in the first activities of the “Rational Unified Process” of software development: business modelling, requirements elicitation, design and analysis [Sommerville 2011]. Business Patterns applied to Conceptual Modelling achieve the reuse of knowledge and experience from previous systems, encapsulated in concrete design and analysis solutions, allowing the validation and verification of functional requirements. They are incorporated to the Conceptual Model together with other patterns that fulfil different stages of modelling, as can be seen in Figure 1.

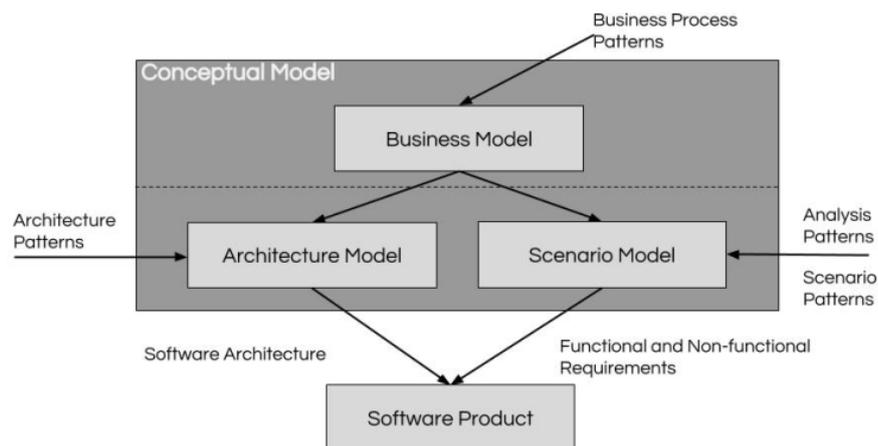


Figure 1: Introduction of Patterns in Conceptual Modelling

The need for a catalogue to classify and group de patterns in relation to their common features arises at this instance, therefore enabling an agile selection when they are needed.

2.2 Business Patterns

Patterns establish generalized solutions that allow to solve common problems in different business situations, can be reused several times, combined and adapted to the context where they are applied [Eriksson and Penker 2000]. Patterns are not

invented nor created; they exist in the models that describe real life business. Thus, they need to be identified and described in a systematized manner. They also specify business vocabularies and can be considered prototypes for production.

Eriksson y Penker distinguish three pattern types, in relation to the problem they address:

- Business Pattern: they tackle organizational domain problems, such as the definition and relation of processes and business rules, organization vision and corporate goals.
- Architectural Patterns: they address de architectural design problems for information systems.
- Design Patterns: they are used in situation where the analysis is already described and focused on producing flexible and adaptable technical solutions.

Business Patterns have the highest level of abstraction, enabling to work with business processes, and are classified in three categories:

- Goal or Objectives Patterns.
- Processes Patterns
- Rules and Resources Patterns.

Goal or Objectives Patterns are models that describe the problem-solution relation with the objectives of a business process. The goal modelling is a critical subject since it affects de global way in which the system will be built. From this point of view, it can be concluded that the processes describe the way to achieve the specified goals with a set of predefined rules and resources.

Processes Patterns are behavioural patterns whose goal is to improve the quality of processes, workflows and other models.

Resources and Rules Patterns provide guidelines for the modelling of rules and resources in the business domain.

Different authors have made several proposals about the basic attributes that patterns need to have, including Gamma, Larman, Fowler, Coplien y Schmidt, amongst others.

The unifying criteria of the attributes template defined by Eriksson and Penker stands out due to its clarity and simplicity. Each Business Pattern is specified using the following "Business Patterns Template":

- Name.
- Intent: general goal of the pattern.
- Motivation: the "forces" that help reach the solution.
- Applicability: problem situations to which the pattern can be applied and this pattern solves.
- Structure: graphic representation of the pattern in standard UML. Class, object, sequence and activities diagrams are allowed.
- Participants.
- Consequences: how this pattern helps to solve the goal defined in Intent, and the problem situations in Applicability
- Example.
- Related Patterns.
- Source /Credit.

Other lines of research are later identified, with processes patterns supported by collaborative efforts of specific standardization. One of them, for example, is Barros who proposes process architecture patterns to design components of a complete system, and invites to use an open pattern development in the Business Model both for public and private organizations [Barros 2019].

Considering de MDD (Model Driven Development) methodology framework that supports them, a software architectural derivation is feasible from the Business Modelling MDD uses diagram models that suffer transformations in the different stages of the construction of a system. “*MDD is the natural evolution of software engineering based in models, enriched by the introduction of automatic transformations between models*”, annotated by [Pons et al. 2010].

There are previous publications [Medina et al. 2018a, Medina et al. 2018b, Medina et al. 2018c, Medina et al. 2018d, Marciszack et al. 2018, Meloni et al. 2019, Medina et al. 2019, Medina et al. 2020] from the authors of this work that develop an analysis model for the definition of patterns in the Conceptual Model of an Information System.

3 Proposal to manage a Business Patterns Catalogue

3.1 Model for defining and choosing Business Patterns.

The result of the stated analysis model for pattern identification and elaboration is a set of patterns classified in different typologies. In the Conceptual Model domain, we refer specifically to Business Patterns, Analysis Patterns, Scenario Patterns and Architectural Patterns.

The scope is focused only on Business Patterns in order to carry out this first approach to pattern cataloguing. Goal - Problem Patterns are the one that stand out amongst Business Patterns, since they define the problem-solution relation of a business process, which they support. Other patterns that complement the functional description for the same business process are also stored: Processes Patterns, and Resources and Rules Patterns. A novel unified graphic is put into consideration in relation to Business Patterns, since it allows an agile visualization of Process Modelling Patterns: basic process structure, interaction process, layer providing process and action workflow [Marciszack et al. 2018].

Besides the “Business Patterns Template” information, previously detailed, keywords are added to introduce the value or importance the pattern has to solve the problems of the business process in which it was implemented. This component is required to develop a standardized and specific thesaurus for the catalogue, and to be more efficient in the text mining functionalities when analysing the initial description of the problem, during search.

Thus, choosing a pattern from a reduced set called catalogue is another one of the critical tasks that must be solved by any Modelling activity that uses patterns.

A method, found in the literature, address the selection of Scenario patterns. Scenarios are formal descriptions of a defined situation inside a domain context.

An alternative for choosing a Scenario Pattern is to use a decision tree with a series of questions about the process, defined by Ridaou, Doorn y Leite heuristics

[Ridao et al. 2000]. They present a predetermined classification of Scenario Patterns and a set of precise and easy to answer questions.

A proposal for a Scenario Patterns Catalogue is done in a later work by [Cortés Bravo et al. 2016], supported on the heuristic by Ridao, Doorn y Leite. It is linked to a concrete study case in the telematics domain, and they state the pros of using scenario patterns as a tool for introduction and improvement of organizational processes.

These kinds of proposals are probably not sufficient support for inexpert designers or for high abstraction level patterns. In this case, data mining techniques can be used to allow the component analysis of patterns inside the catalogue and, with interaction with the designer, suggest the best-fit pattern for the problem. Bouassida presents an interactive method for pattern selection, implemented in a Design Patterns catalogue [Bouassida et al. 2015]. Paz Menvielle also proposes searching patterns in free text writing, inside a limited domain, represented in a directed graph database. [Paz Menvielle et al. 2018].

This work also starts from the natural language analysis in the problem description made by the designer. The search for keywords inside that description is carried out first during the business modelling, and can be applied early in Conceptual Modelling. The next step consists in finding an Objective Pattern with associated keywords for the system that we are Modelling, since problems solved by the process are described in this type of pattern. Later on the Business Processes Pattern is obtained in a unified graph, proposed by [Marciszack et al. 2018], which develops the relation between the involved processes and how the resources change between them. The specification of this pattern's main process may have other kinds of associated patterns, such as Analysis Patterns and Scenario Patterns that complement the functional specification of software requirements, as shown in Figure 1.

3.2 Pattern Catalogue Prototyping

“PatCat” (Pattern Catalogue) is a web catalogue of Business Patterns applied to successful implementations of Electronic Government. The application is developed under the scope of the academic research project called “Analysis model for application of Good Practices Patterns in Conceptual Modelling of Electronic Government” [Medina et al. 2018b]. The project's goal is the definition of good practices patterns for E-Government than can be used Conceptual Modelling of public organizations web sites, in order to optimize some software quality dimensions during their construction, such as reusability.

In the same way that other information systems solve well defined problems, it would be advisable that a set of best practices for Conceptual Modelling of digital citizenship portals is made available, which could be referred to and leveraged as a baseline for requirements specification at the starting phase.

The “PatCat” catalogue allows registering Business Patterns grouped by pattern type and business process. The initial planned functionalities are reduced in order to make the tool available to allow experimenting with this proposal, following the guidelines of Evidence Based Software Engineering [Wholin et al. 2012, Genero Bocco et al. 2014].

It is important to say that future software versions must include all functionalities to withstand the maximum generalization level.

Figure 2 shows the Electronic Government pattern screen from a selected process, and the input of the general description of the public problematic to be solved:

Figure 2: “PatCat” search screen

For example, for a business process catalogued as “Work Safety Monitoring”, searching terms such as “work risk” or “safe environment” could appear inside the problem domain description. These keywords must be associated in the Business Pattern Catalogue, especially in the Objective Patterns, so that they can be chosen.

From now on, the vector model used to develop the search algorithm for the catalogue is exposed.

3.3 Pattern Search Engine Architecture

The main idea of the data model for the search engine is that every document stored in the database, in this case catalogue patterns, is represented as a vector in a k -dimensional vector space. Each coordinate is a number, which indicates the value of a term identified as keyword in the query, for each document in the catalogue. This entry values are translated to input variables of a mathematical model that allows to compute the relevance of a document for a given query. The value assigned to each document for a given query allows creating a relevance order.

This sorting and assessment are solved with an inverted indexes structure constituted by two elements:

- Vocabulary: set of every unique term registered in the database.
- Publication list: the list of documents linked to queried terms and their relevance value, for every existing term.

When both structures were obtained, a simple query solves the search providing the most relevant documents as answer.

3.4 Technical Specifications of “PatCat” tool

“PatCat” tool is constituted by the following technical dimensions: Front-end, Back-end and a Database.

- Front-end: The web application structure answers the HTML5 standard, along with the Pug template engine, that enables a faster and simpler coding for view optimization. On the other hand, web pages execute Javascript code with Ajax and CSS when published, to provide styling and formatting to user interfaces. The text editor used is “Visual Studio Code”, which is prepared to build and debug web and cloud applications.
- Back-end: Developed in Node.js programming language, combined with Express.js, a fast, minimalistic and flexible web infrastructure for Node.js (see Figure 3). This plugin provides a solid API, with utilities and methods for HTTP publishing and middleware for web and mobile app development.

```
Schema = mongoose.Schema,
appPattSchema = new Schema ({
  appPattName: { type: String },
  appPattImg: { type: String },
  appPattBody: Schema.Types.Mixed
}),
{
  collection: 'appPatterns'
}),
appCasesSchema = new Schema ({
  appGovProc: { type: String },
  appTitle: { type: String },
  appPatterns: [appPattSchema],
  appPaper: { type: String },
  appDescription: { type: String }
}),
{
  collection: 'appCases'
}),
```

Figure 3: Back-end code snippet

- Database: A document-based, distributed and general-purpose database, MongoDB, is used. This database was created for web, mobile and cloud development. It offers a powerful query language, rich and expressive, that allows filtering and sorting by any attribute, regardless of how deeply nested it could be inside a document. It is compatible with aggregations and other use cases such as advanced text query.
- Class diagram: Business processes, classified by category, are registered at a higher domain level (see Figure 4). For example, a category could be Government Educational System and matriculation to an educational

institution could be one of its processes. On the other hand, patterns are grouped by type of pattern, such as Business Pattern, Analysis Pattern, etc. The type of pattern is a recursive class which enables hierarchical grouping, such as an Objectives Pattern is a subtype of a Business Pattern. Publishing a pattern to the catalogue establishes a relation between a business process and a defined pattern. Furthermore, it has search words or terms associated valued by frequency. Vocabulary classes, as explained in the data base description, manage terms. Finally, the elaboration and diffusion in academic and industry fields is supported by the referred scientific publications in each catalogued pattern.

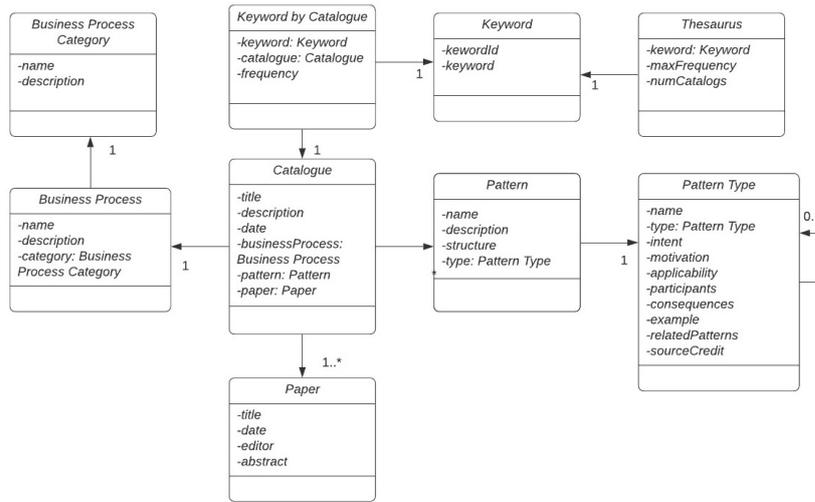


Figure 4: “PatCat” class diagram

4 Study case of Business Patterns applied to Public Software Modelling

To test “PatCat” with a real example, Objective Patterns were catalogued in the Business Modelling for an academic management system of Cordoba Government: the State Educational System.

The selected business process for this test is the “Inscription to a school or institution”, where only Objective Patterns are modelled.

The Education System of Cordoba Province is a system of considerable complexity, constituted by around 5000 public and private schools, and 900.000 students approximately. It is structured in four educational levels: Initial, Elementary, Mid and Superior, and four different types of education: Common, Special, Adults and Technical. The combination of educational levels and types is detailed in Table 1:

Levels	Educational types			
	Common	Special	Adults	Technical
Initial	X	X		
Elementary	X	X	X	
Mid	X		X	X
High	X			X

Table 1: Relationship between Educational levels and types

Inscription to a school or institute is the process of registering a person as student in a school or institute. This definition applies to every educational level, with some exceptions. A person can enrol only once in the superior level to a school and career and in the other levels must enrol every school year. Cordoba Province only allows a student to enrol in one school, and at the superior level, he/she can enrol in more than one school, since a student can do more than one career and can assist to different institutions.

The initial level offers up to three study years called classrooms. This can be three years, four years and five years. Elementary and mid are constituted by six years of study called grades or years respectively. Furthermore, a school can offer more than one course for each year of study, which are called divisions and symbolized with letters: A, B, C, etc. Such as, second grade A or first year B. Few schools have the complete offer and, in addition, there can be more than one classroom for every year and, like in previous levels, letters identify divisions.

Mid schools are constituted by two cycles: basic cycle and specialization cycle, with three years each. The basic cycle is common to every schools of the same level, but there are some extra subjects in private and rural schools. Most Mid schools with technical specialization have four years in the specialization cycle.

The enrolment in superior level is addressed as the inscription to a career, this is unique and a student can have more than one enrolment in the case that he/she wants to do more than one career. Figure 5 shows the general enrolment process for the Educational System of Cordoba Province:

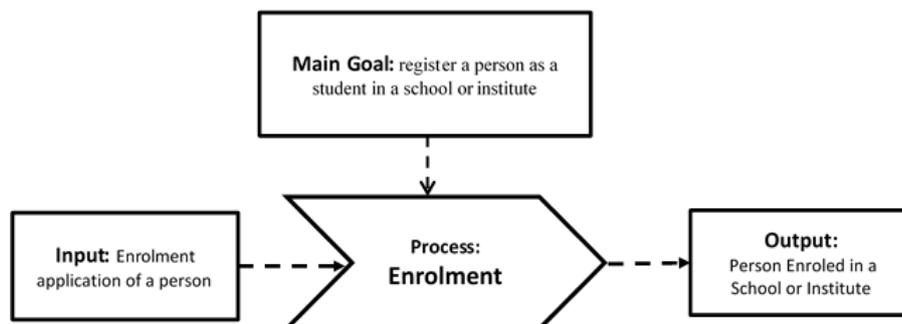


Figure 5: Enrolment process diagram

4.1 Objectives Pattern applied to the process of Enrolment to a School or Institute.

To model objectives in business processes there is the Objectives Pattern that is made up of three other patterns:

- Objectives Decomposition Pattern.
- Objectives Assignment Pattern.
- Objective - Problem Pattern.

These patterns are generally applied in the presented order and, obviously, are interrelated.

Elementary School with common education type is selected among the available educational levels and types in the Government Educational System (see Table 1), and the patterns in this example will be applied to it.

4.2 Objectives Patterns Application

Every student of elementary school with common education type is enrolled in a grade and a division, such as second grade A. The division, as previously stated, is used to identify more than one grade of a same level. There is a uniform syllabus for every "Elementary Level" with common educational type of Cordoba Province. Besides assigning students to a grade or division, personal data from students is registered, as well as their tutor's data, since they are minors and need to have a person that responds for the student and the required documentation.

The first step is to apply the Objective Decomposition Pattern, then de Objective Assignment Pattern and finally de Objective - Problem Pattern.

4.3 Objectives Decomposition Pattern

Starting from de global objective and the intrinsic features of each educational level and type, we can elaborate the following pattern, shown in Figure 6:

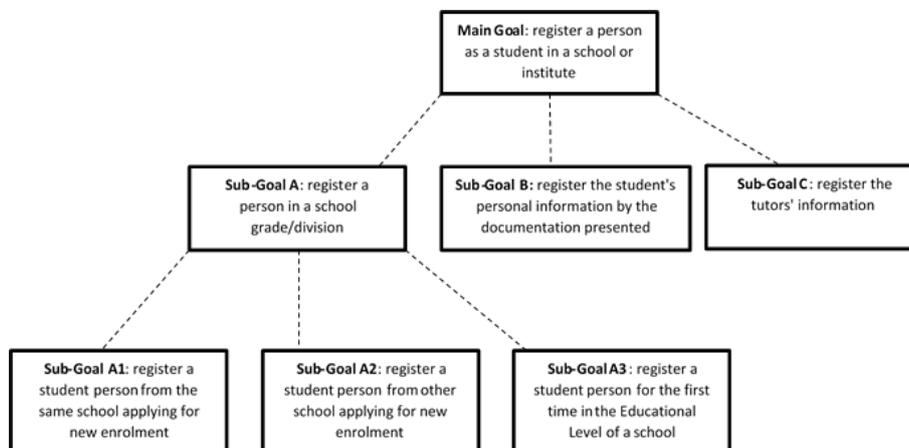


Figure 6: Objectives Decomposition Pattern of the Enrolment process on Elementary level with Common Educational type

4.4 Objective Assignment Pattern

The result of the aforementioned analysis model for pattern identification and elaboration is a set of patterns classified in different typologies. We refer specifically to Business Patterns, Analysis Patterns, Scenario Patterns and Architectural Patterns in the domain of Conceptual Modelling

Three sub-processes can be identified according to the application of the Objective Assignment Pattern:

- Sub-Goal A: Enrolment Management Sub-Process.
- Sub-Goal B: Personal Data Management Sub-Process.
- Sub-Goal C: Tutor Management Sub-Process.

4.5 Objective - Problem Pattern

When applying this pattern to the stated goals we found different problematic situations. Figure 7 presents an application example of the A3 Sub-Goal:

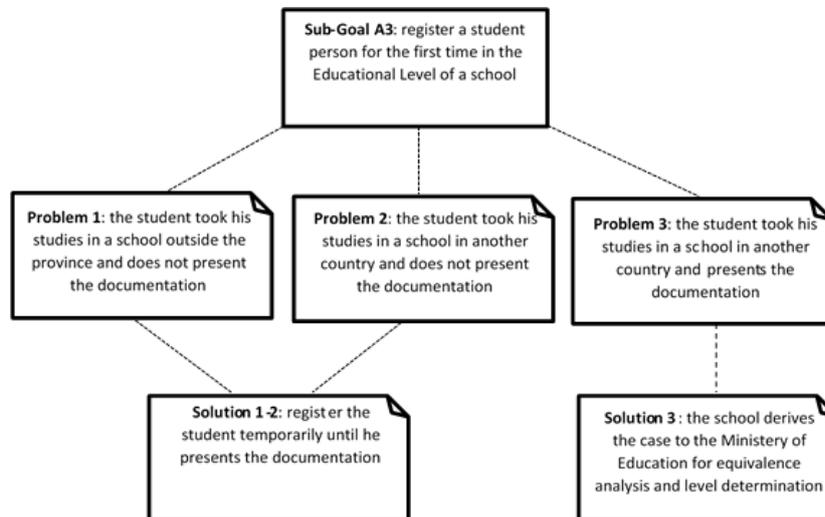


Figure 7: Objective – Problem Pattern applied to Sub-Goal A3

The Enrolment process for Elementary school with common education type incorporates all the processes obtained by applying the defined Objective Patterns.

The graph in Figure 7 shows the main process and its decomposition in Sub-Process related to the Problem - Objective Pattern of the A3 Sub-Goal:

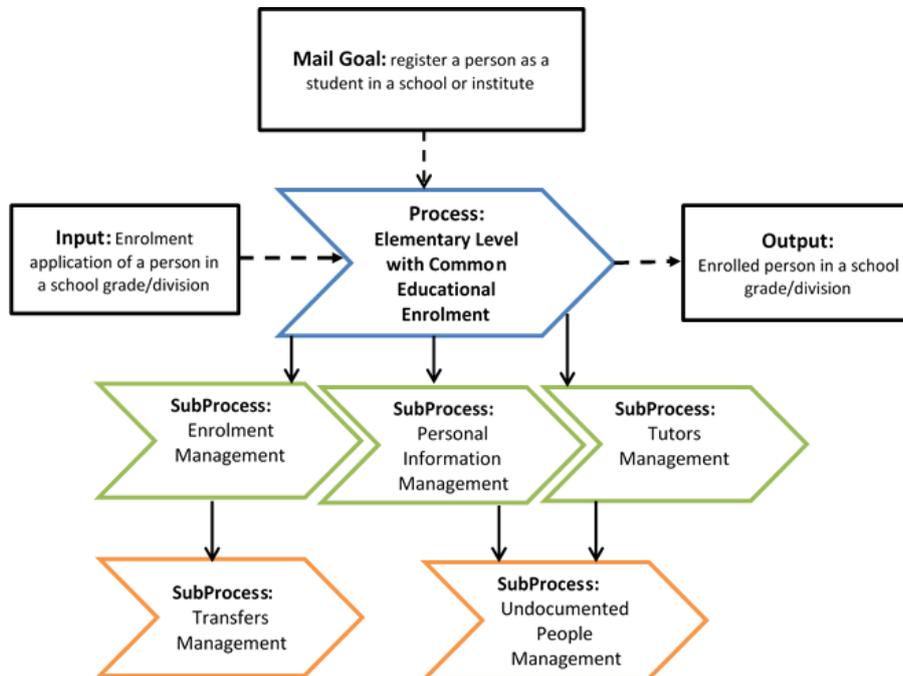


Figure 8: Business Process Pattern of Elementary level with Common Educational type

4.6 Keywords and pattern selection

The Objective Decomposition Pattern, such as the one shown in Figure 6, is useful to determine keywords and assign a relevance value for the pattern when there are no automatized text mining methods.

Terms like enrolment, school, institute and student can be assigned a value of $n = 10$ in this way, since they are characteristic of the first level of the pattern, second level terms can be valued with a $n - 1$ weight, and so on. In the application thesaurus, each registered keyword for the catalogue will have the maximum value and the total quantity of patterns associated with it, which additionally allows to measure the importance of the term in the context of the whole catalogue.

Lastly, the selection of the pattern is carried out by interaction with the application interface, which is shown in Figure 2, by introducing the summary of the Electronic Government process to model. The pattern selection is successful when the keywords can be identified, exist in the catalogue and are associated to pertinent patterns.

5 Results

In the previous section, a practical case of Business Patterns application is presented where the “PatCat” tool was used and Objective Patterns were catalogued in the Business Modelling of an academic management system of Córdoba Government: The State Educational System [Meloni et al. 2019].

This system presents different educational levels and types that make up the selected State Educational System (see Table 1). The Elementary level with a common type of education was chosen to apply the patterns in this study case. The Objective Pattern that was obtained, and that is presented in Figures 6, 7 and 8, it is applicable to four combinations of educational levels and types of the State Educational System of Córdoba Government:

- Initial Level with Common Type.
- Elementary Level with Common Type.
- Mid-Level with Common Type.
- Mid-Level with Technical Type.

In addition to the aforementioned, E-Gov software patterns were catalogued in two other domains:

- A Business Pattern and an Objective Pattern of a system of citizens inquiries to the public pension agency of Córdoba Government [Medina et al. 2019].
- A Business Pattern, an Objective Pattern and three Architecture Patterns of a mobile software that supports the occupational safety process of a public self-governed entity, whose main activity is to supply electric energy to Córdoba province [Medina et al. 2018d, Medina et al. 2020].

As a result of these first experiences, it was reliably verified that the reuse of Good E-Gov Good Practices is feasible using Business Patterns, Objective Patterns and Architecture Patterns. In this way, the purpose of every pattern was fulfilled: to obtain the solution models to be reused in information systems, with the same problem domain, adapting them to other contexts.

Through new replications of this experiment, more statistical data can be obtained and evaluated, which will allow establishing the range of conditions in which the proposed hypothesis continues to be fulfilled, without underestimating the quality of the data obtained with the current one, which has been tested. And, that it will collaborate incrementally in the selection of the presented “PatCat” tool.

6 Conclusions and Future Work

Objective Patterns application in software Modelling allows to clarify the requirements elicitation, lay out problems, find solutions and similarities between the scenarios to simplify the process modelling, as can be seen in the example presented in section 4. These patterns show further advantages when introduced at the beginning of the analysis, seeking consensus on the objectives to accomplish with the process owner. This early activity and its impact in the sustainability of the software product is often overlooked.

The present work explains the first stage of a Business Pattern Catalogue, looking to reuse them in the Conceptual Model of informatic systems. The “PatCat” catalogue

allows registering Business Patterns which can be grouped by pattern type and business pattern. In this initial version, functionality has been reduced in order to obtain an experimental tool to test its usability and gather conclusions. Future versions of the software must include full functionality to withstand the maximum level of generalization. It is necessary to have a pattern management and selection model that allows to manage them systematically and promotes reuse so that the implementation of this proposal is feasible.

Pattern introduction is not time consuming, comparing it to the valuable insights obtained, and has an intrinsic ease of use that allows an agile introduction to analysis and design activities. It is an advisable good practice in business processes studies. Reusability is a desirable software quality dimension to optimize when building a new system. Therefore, a specific application to manage a pattern catalogue is useful to standardize and speed up this software design task.

The goal of a pattern catalogue is to have solution models to be reused in similar processes, being helpful in really big or complex systems. With this in mind, the catalogue must facilitate different search ways according to the features of the pattern types it manages. For Objective Patterns, it has to include the basic attributes of Objective Decomposition Patterns, such as Objective Assignment Patterns and Objective - Problem Patterns, and the ability to relate them to Business Processes. Therefore, it is necessary to offer to query the catalogue by a pattern type, by a process or by both, or by business processes, among other patterns attributes.

We still have to solve the management of synonyms or derived terms with the same meaning, and the relevance computation for each one of them, in the same way that is done when assigning them to a pattern.

“PatCat” has now gone through the first stage of construction of a Business Patterns catalogue, thus, the future work is to progress in the construction of that catalogue, extending the scope to other pattern types, and optimizing the search through a pattern selection method with data mining and natural language analysis functionalities.

State of the art for Business Patterns has not identified a methodology for the analysis and construction of a catalogue to support this work, which is put to consideration as a first approach, using as catalogue publications and experiences for other pattern types as references.

References

- [Alexander, 77] Alexander, C.: A pattern language. Oxford University Press, New York, 1977.
- [Alexander, 79] Alexander, C.: The Timeless Way of Building. Oxford University Press, New York, 1979.
- [Barros, 19] Barros, O.: A process architecture pattern and its application to designing health services: emergency case, *Business Process Management Journal*, 26, 2, 513-527, 2019.
- [Bouassida et al., 15] Bouassida, N., Jamoussi, S., Msaed, A., Hanène Ben-Abdallah, H.: An Interactive Design Pattern Selection Method, *J.UCS (Journal of Universal Computer Science)*, 21, 13, 1746-1766, 2015.

- [CLAD, 15] CLAD Centro Latinoamericano de Administración para el Desarrollo: Cartas Iberoamericanas del CLAD, Secretaría de la Función Pública de la República del Paraguay, 2015.
- [Cortés Bravo et al., 16] Cortés Bravo, C.A., Abud Figueroa, M.A., Romero Torres, C., & Peláez Camarena, G.: Propuesta de un Catálogo de Patrones de Escenario para la Definición de Requisitos, *ReCIBE Revista electrónica de Computación, Informática, Biomédica y Electrónica*, Año 5, 1, 52, 2016.
- [Eriksson and Penker, 00] Eriksson, H.-E. & Penker, M.: *Business Modeling with UML: Business Patterns at Work*, OMG Press, 2000.
- [Estrada et al., 18] Estrada, E., Maciel, R., Peña Pérez Negrón, A., López Lara, G., Larios, V., Ochoa, A.: Framework for the Analysis of Smart Cities Models, in Mejia, J. et al. (Ed.): *Trends and Applications in Software Engineering. Proceedings of the 7th International Conference on Software Process Improvement (CIMPS 2018)*, 261–269, Springer, 2018.
- [Gamma, 94] Gamma, E., Helm, R., Johnson, R., Vlissides, J.: *Design Pattern - Elements of Reusable Object-Oriented Software*. Addison Wesley, 1994.
- [Genero Bocco et al., 14] Genero Bocco, M., Cruz-Lemus J.A., Piattini Velthuis, M.G.: *Métodos de investigación en ingeniería de software*, Ed. Ra-ma, 2014.
- [Marciszack et al., 18] Marciszack, M.M., Moreno, J.C., Sánchez, C.E., Medina, O.C., Delgado, A.F., Castro, C.S.: *Patrones en la construcción del Modelo Conceptual para sistemas de información*, Editorial edUTecNe, Universidad Tecnológica Nacional, 2018.
- [Medina et al., 18a] Medina, O.C., Marciszack, M.M., Groppo, M.A.: Aproximación Descriptiva a las Buenas Prácticas de Gobierno Electrónico y a su incorporación en el Modelado Conceptual de Sitios Web Públicos de Argentina, *Revista Tecnología y Ciencia*, 31, 99-110, Universidad Tecnológica Nacional, 2018.
- [Medina et al., 18b] Medina, O.C., Marciszack, M.M., Groppo, M.A.: Un Modelo de Análisis para aplicación de patrones de Buenas Prácticas en el Modelado Conceptual de Gobierno Electrónico, in proceedings of WICC 2018, Red UNCI y UNNE Universidad Nacional del Nordeste, 2018.
- [Medina et al., 18c] Medina, O.C., Marciszack, M.M., Groppo, M.A.: Proposal for the patterns definition based on good practices for the electronic government systems development, in proceedings of CISTI 2018 – 13th Iberian Conference on Information Systems and Technologies, 2018.
- [Medina et al., 18d] Medina, O.C., Pérez Cota, M., Marciszack, M.M., Martin, S.M., Pérez, N., Dean, D.D.: Conceptual Modeling of a Mobile App for Occupational Safety Using Process and Objectives Patterns, in *Trends and Applications in Software Engineering. Proceedings of the 7th International Conference on Software Process Improvement (CIMPS 2017)*, 186-195, Springer, 2018.
- [Medina et al., 18e] Medina, O.C., Cánepa, P.A., Gruppo, M.O., Groppo, M.A.: Un caso de estudio de patrones de Gobierno Electrónico para gestión de consultas de ciudadanos, in proceedings of CONAIIISI 2018, Red RIISIC, CONFEDI y Universidad CAECE, 2018.
- [Medina et al., 19] Medina, O.C., Groppo, M.A., Marciszack, M.M.: Una aproximación a la definición de Patrones a partir de Buenas Prácticas para el desarrollo de sistemas de Gobierno Electrónico, in *Actas de Jornadas y Eventos Académicos de UTN (AJEA)*, 4, 274-278, Universidad Tecnológica Nacional, 2019.

- [Medina et al., 20] Medina, O.C., Romero, M.s., Romero, R.A., Martin, S.M., Marciszack, M.M.: Using Architecture Patterns in the Conceptual Model of an eGov Software, in Trends and Applications in Information Technology and Systems. Proceedings of the 2020 International Conference on Information Technology & Systems (ICITS 2020), 54-63, Springer, 2020.
- [Meloni et al., 19] Meloni, B.E., Pérez Cota, M., Medina, O.C., Marciszack, M.M.: Objectives Patterns Applied to the Business Model of a Public Education System, in Trends and Applications in Software Engineering. Proceedings of the 8th International Conference on Software Process Improvement (CIMPS 2018), 13-22, Springer, 2019.
- [Paz Menvielle et al., 18] Paz Menvielle, M.A., Corso, C.L., Guzmán, A., Casatti, M.G., Ligorria, K.: Búsqueda de patrones en un dominio representado en una base de datos de grafos dirigidos, in proceedings of CONAIIISI 2018, Red RIISIC, CONFEDI y Universidad CAECE, 2018.
- [Pons et al., 10] Pons C., Giandini R., Pérez G.: Desarrollo de Software dirigido por Modelos (Conceptos teóricos y su aplicación práctica), Edulp - Editorial de la Universidad de la Plata, 2010.
- [Ridao et al., 00] Ridao, M., Doorn, J., Leite, J.C.S.P.: Uso de Patrones en la Construcción de Escenarios, in Anais WER 2000, III Workshop em Engenharia de Requisitos, Rio de Janeiro, Brasil, 2000.
- [Rosales-Morales et al., 20] Rosales-Morales V.Y., Sánchez-Morales L.N., Alor-Hernández G., García-Alcaraz J.L., Sánchez-Cervantes J.L., Rodríguez-Mazahua L.: *ImagIngDev*: A New Approach for Developing Automatic Cross-Platform Mobile Applications Using Image Processing Techniques. *The Computer Journal*, V. 63, I. 5, 2020
- [Sommerville, 11] Sommerville, I.: *Ingeniería de Software 9a Edición en español*, Pearson, 2011.
- [Varela Rey, 16] Varela Rey, A.: Beneficios del intercambio de buenas prácticas municipales, *INNOTEC Gestión*, 7, 55-59, Laboratorio Tecnológico del Uruguay, 2016.
- [Velasco-Elizondo et al., 16] Velasco-Elizondo P., Marín-Piña R., Vázquez-Reyes S., Mora-Soto, A., Mejía-Miranda, J.: Knowledge representation and information extraction for analysing architectural patterns. *Science of Computer Programming*, V. 121, Elsevier, 2016
- [Vo and Lai, 14] Vo, G.N., Lai, R.: A Secure Multi-Layer e-Document Method for Improving e-Government Processes, *J UCS (Journal of Universal Computer Science)*, 20, 11, 1583-1604, 2014.
- [Wholin et al., 12] Wohlin, C., Runeson, P., Höst, M., Ohlsson, M., Regnell, B.: *Experimentation in software engineering*, Springer, 2012.