Universidad Carlos III de Madrid
www.uc3m.es

TESIS DOCTORAL

A Design Pattern Language to Assist the Design of Alarm Visualizations for Operating Control Systems

Autor:

Rosa María Romero Gómez

Director:

Prof. Dr. Paloma Díaz Pérez

Departamento de Informática

Leganés (Madrid), Noviembre, 2015

TESIS DOCTORAL

A Design Pattern Language to Assist the Design of Alarm Visualizations for Operating Control Systems

Autor: Rosa María Romero Gómez

Director/es: Prof. Dr. Paloma Díaz Pérez

Firma del Tribunal Calificador:

Firma

Presidente: (Nombre y apellidos)

Vocal: (Nombre y apellidos)

Secretario: (Nombre y apellidos)

Calificación:

Leganés, de de

A mis padres y a mi hermana

Resumen

Con la creciente relevancia de la visualización como mecanismo para el análisis y exploración de grandes y complejos volúmenes de datos, la investigación en visualización ha puesto de manifiesto la necesidad de reutilizar conocimiento previo de diseño en lugar de comenzar desde cero. Esta necesidad es especialmente importante en el diseño de sistemas de control, en los cuales las visualizaciones de alarmas se han convertido en artefactos clave para los operadores humanos a la hora de adquirir una consciencia del estado del proceso controlado. En este contexto, los diseñadores de visualizaciones de alarmas tienen que hacer frente a dos desafíos de diseño recurrentes y fundamentales relacionados con cuestiones de escalabilidad visual y el proceso de dar sentido a la información. En particular, enfrentarse a tales desafíos por parte de estos diseñadores requiere poseer conocimiento de diseño relacionado con diversas áreas de conocimiento tales como la Gestión de Alarmas, Factores Humanos y el Diseño de Visualización. Sin embargo, es difícil que un diseñador sea un experto en cada una de las áreas de conocimiento involucradas en un problema de diseño, lo cuál requiere años de experiencia. Ante esta situación, un enfoque adecuado a seguir es la reutilización de conocimiento de diseño previo, lo cual posibilita la amplificación de las capacidades de los diseñadores a la hora de generar diseños. No obstante, los actuales mecanismos existentes para la reutilización de conocimiento previo de diseño de visualizaciones de alarmas son demasiado abstractos, no son lo suficientemente exhaustivos en la cobertura de factores claves para el diseño de visualizaciones de alarmas, ni se encuentran adecuadamente acoplados, lo cual dificulta su utilización por parte de diseñadores no experimentados.

En este trabajo de investigación, tales limitaciones han sido cubiertas mediante el desarrollo de un lenguaje de patrones de diseño para visualizaciones de alarmas. Un lenguaje de patrones de diseño puede caracterizarse como un enfoque adecuado de reutilización de conocimiento de diseño previo que facilita su diseminación a diseñadores no experimentados. Derivado de una revisión de modelos descriptivos y reglas de diseño para la Gestión de Alarmas, Factores Humanos y el Diseño de Visualización, así como de la realización de dos casos de estudio enmarcados en el contexto de dos proyectos de investigación, Energos y Emercien, el presente trabajo de investigación describe el espacio de diseño de visualizaciones de alarmas. A continuación, guíado por dicha caracterización del espacio de diseño y como resultado de una revisión extensiva de recursos relevantes pertenecientes a las anteriormente mencionadas áreas de conocimiento, este trabajo de investigación sistematiza el conocimiento de diseño reutilizable para el diseño de visualizaciones de alarmas a través de la definición del lenguaje de patrones de diseño.

La estructura del lenguaje de patrones de diseño ha sido analizada de manera analítica y sus elementos constitutivos, los patrones de diseño, han sido evaluados por diseñadores expertos. Asimismo, este lenguaje de patrones se ha aplicado en diferentes contextos, lo cuál demuestra su factibilidad de uso para crear visualizaciones de alarmas en diversos dominios. Finalmente, este lenguaje de patrones ha sido utilizado por diseñadores no experimentados. Su utilización por parte de los mismos demuestra la utilidad del lenguaje de patrones para proveer un acceso sencillo al cuerpo de conocimiento de diseño existente sobre visualizaciones de alarmas y abre futuras líneas de investigación de interés.

Abstract

With the growing emphasis on visualization as a mechanism for analysing and exploring large and complex data sets, visualization research has recognized the need of reusing prior design knowledge instead of starting from scratch. This fact is especially relevant in designing control systems in which alarm visualizations are key artefacts for human operators to maintain an awareness of the state of the process under control. In this context, designers are required to face two fundamental recurrent design challenges related to visual scalability and sense making issues, which involve having design knowledge from different knowledge areas including Alarm Management, Human Factors, and Visualization Design. Nevertheless, no single designer can be an expert in every relevant knowledge area, and becoming proficient may require years of experience. One relevant approach to assist in such multi-dimensional design process is to reuse prior design knowledge, which supports and amplifies designers' abilities of design generation. However, existing design knowledge reuse approaches for alarm visualization design can be too abstract, not comprehensive enough, and loosely coupled, being difficult to be applied by non-experienced designers. In this research work, such limitations are addressed by developing a design pattern language as a fitting design knowledge reuse approach to disseminate reusable alarm visualization design knowledge. Derived from both a review of descriptive models and design rules for Alarm Management, Human Factors, and Visualization Design and two case studies framed within the context of two different projects, Energos and Emercien, this research work describes the design space for alarm visualization design. Then, taking such characterization of the design space into account and after an extensive review of relevant sources in the aforementioned knowledge areas, this research work systematizes reusable design knowledge for alarm visualization design through the definition of the design pattern language.

The structure and elements of this design pattern language have been, respectively, analytically analized and evaluated by expert designers. This design pattern language has been also applied in different contexts. It demonstrates the feasibility of this design pattern language to be used across application domains. Finally, it has been sucessfully used by non-experienced designers. It demonstrates the utility of the design pattern language to provide non-experienced designers with an easy access to the existing body of knowledge of recognized alarm visualization design solutions for operating control systems and open up interesting lines for future research.

Contents

List of Tables

List of Figures

Chapter 1. Introduction

There is not such thing as information overload. There is only a bad design.

- *Edward Tufte,* American statistician

A control system is defined as *"a device, or set of devices, that manages, commands, directs or regulates the behaviour of other device(s) or system(s)"* [119]. Control systems have been used over time to solve problems of practical importance with enormous impact on society. As a consequence, control systems have proliferated enormously. They can be found within a range of industrial domains, including *electric power grids*, *transportation networks*, or *water management systems*. Lately, the intensive use of information technologies has resulted in a proliferation of these control systems in other domains such as *emergency response* [82][81]; in particular, in case of natural disasters including earthquakes or floods. In current control systems, equipment is separated in functional areas and is installed in different work areas of a controlled process. The human operator monitors and manipulates the set points of the process parameter from a central control room. In particular, he/she visualizes the process information transmitted from the process area and displayed on the computer terminal through information displays. Computer-based information displays provide the capability to human operators to process data of controlled processes through the use of representation methods, such as graphics and integrated displays. With the help of computer-based information displays, human operators decide on the actions required for controlling the process.

The design of computer-based information displays determines what should be displayed, how it should be displayed, and how to interact with these displays. The way the information is displayed may result from the consideration of both the physical surroundings in which the operator is required to work, a *control room*, and the nature of human operator tasks. Owing to the high complexity of current control systems, where many parameters on many different locations need to be supervised and controlled, the execution of human operator tasks can be characterized under the "operation-by exception" approach. In keeping with the research literature on process control, human operators can perform their tasks according to either the management-by-awareness approach [75] or the operation-by exception approach [31][149]. The management-by-awareness approach characterizes the human operators' activity as a continuous monitoring of the system to try to anticipate and avoid problems' occurrence. On the contrary, the operator-by-exception approach describes the human operators' activity as triggered into action by an upcoming cue or *alarm* [31], instead of focusing on monitoring any feature or parameter of the system; this approach has been more specifically conceptualized under the notion of "alarm-initiated activities" [125]. This notion characterizes human operator tasks as focused on the use of alarms. Aiming at supporting human operators to maintain an awareness of the state of the controlled process, alarm visualizations have become key computer-based information displays for operating control systems [87][112]. Alarm visualization refers to as *"the visual method(s) by which alarm coding and messages are presented to control room operators"* [64]. The design of alarm visualizations is not a trivial process.

Designing alarm visualizations for operating control systems requires designers to face multiple times two fundamental design challenges: (1) *how to display large volumes of alarms with multiple attributes such as typology, priority or location,* and (2) *how to assist human operators in the process of making sense of these large volumes of alarm information in order to get to know the state of the process being controlled*. These design challenges are related, respectively, to the concepts of *visual scalability* [41] and *sense making* [70], which involve understanding different factors affecting them related to specialized knowledge areas such as Alarm Management, Human Factors, and Visualization Design. Nevertheless, no single designer can be an expert in every relevant knowledge area, and becoming proficient may require years of experience [51]. Moreover, in today's global and competitive business environment, designers are under increasing pressure to perform better in terms of low-time, high-quality and high value output that can provide competitive advantage for the organisation [84]. One relevant approach to assist in such design process is to *reuse prior design knowledge*, which supports and amplifies designers' abilities of design generation. Prior design knowledge is defined as "*the conceptual ideas, lessons, and representations captured in the design artifacts created or collected when solving a design problem"* [84]. This definition characterizes the design process as an activity that generates knowledge, and implicitly this knowledge can be reused. In alarm visualization design, assisting designers in the process of reusing prior design knowledge versus starting from *scratch* can help to improve both the efficiency and effectiveness of the design process.

1.1 Research Problem

Alarm visualization design can be defined as *"the process of creating the visual representations by which alarm coding and messages are presented to control room operators".* In order to support the reuse of prior design knowledge, there exists different design rules, including design principles, guidelines, standards, and visual languages from different knowledge areas including Alarm Management, Human Factors, and Visualization Design, which document the expectations and best practices for effective alarm visualization design.

In particular, within the Alarm Management area, several industrial manufacturers, and vendors from worldwide have defined a number of guidelines [16] and standards [40][64] to facilitate the design of alarm visualizations based on several studies conducted with operators. What these design rules highly recommend to designers is the achievement of a balance between the need for content and simplicity of presentation. However, they describe just the minimum acceptable for achieving this balance. They focus on *what to do* rather than *how to do it.* It is designers' task therefore to establish how these rules are applied to the design of alarm visualizations. Moreover, it is also designers' task to ascertain how these rules should be combined in order to generate a complete alarm visualization design.

Similarly, within the Human Factors area, a set of design principles for designing alarm visualizations for Situation Awareness (SA) [44] has been defined. SA can be characterized as the most relevant goal for human operators in handling alarm information [44]. It refers to knowing what is happening around the operation and what the information means at this moment and in the future. According to that, these design principles are based on the Endsley's conceptualization of SA as a knowledge state. They provide a guidance to make a design of alarm visualizations that makes it possible to achieve a knowledge state. However, they leave aside the support to the process of achieving this knowledge state, which have been conceptualized as the *sense-making* process. This process can be characterized as a key design factor while designing control systems across application domains [55]. Moreover, as in the case of the design rules from the Alarm Management area, these design principles are too abstract and their specific implementation will depend on the circumstances and the designer's creativity.

Within the Visualization area, diverse design principles [94][135] and visual languages [83][144] from diverse fields such as cognitive science and cartography have been defined. These design rules, respectively, allow designers to understand how visualization means enable human cognition and to identify the basic units of a visualization. Nevertheless, most of them do not take into account the support of visualizations to specific data types, volumes of information, display sizes, and tasks performed by the user. They assume an absolute validity across design situations while usually they lack describing them. A potential result of that may be bad design choices to the data types and tasks at hand or misinterpretations by designers.

In summary, existing design rules for alarm visualization design show some limitations to be applied by *non-experienced designers*, who lack professional or specialized knowledge in the domain. In particular, there are three main problems: (1) these design rules **are not comprehensive enough:** they provide a partial consideration of the required design factors for alarm visualization design; (2) these design rules **are too abstract:** they provide a too high-level description of the important features of alarm visualizations, but they do not express how to achieve them; (3) these design rules are **loosely coupled**: rhey do not provide any description of how these design rules should be combined together in order to generate alarm visualizations.

1.2 Research Aim & Objectives

In order to address the research problem as stated above, this research work is aimed **at assisting non-experienced designers in the process of reusing previous design knowledge for designing alarm visualizations for operating control systems.** The approach of this work then is to assist such reuse process by addressing three main objectives (**O**):

- **O1.** Capturing reusable design knowledge from different knowledge areas related to recurrent design challenges for alarm visualization design.
- **O2.** Providing multiple ways to organize such reusable design knowledge in order to assist designers along different stages of the design process.

- **O3.** Combining such reusable design knowledge in a cohesive way to assist non-experienced designers.

These objectives are taken as the input for the design and development of a suitable solution. In particular, each objective represents a specific aspect that the proposed solution satisfies.

1.3 Research Questions

According to the research aim and objectives of this research work, the following research questions (RQ) have been formulated:

- **RQ 1.** *What is the design space for alarm visualization design?* This research question addresses the characterization of recurrent design challenges for alarm visualization design in terms of factors affecting them and knowledge areas involved.
- **RQ 2.** *How to support non-experienced designers in the process of reuse previous design knowledge for designing alarm visualizations?* This research question addresses not only the selection of an appropriate design knowledge reuse approach for alarm visualization design but also the collection, the organization in different forms and the combination of reusable design knowledge for alarm visualization design in a more cohesive way in order to assist non-experienced designers.

1.4 Research Methodology

The visualization design discipline, a sub-discipline of Human-Computer Interaction (HCI), has a strong tradition of design-oriented research, where the design guidance for visualization environments, the presentation of a designed system to illustrate some new technique or visualization capability, or the evaluation of such visualization capability are considered a meaningful and important scholarly contribution [100]. Following this tradition, this research work particularly contributes to the design guidance of alarm visualizations for operating control systems. To achieve that, it applies the research methodology proposed by Offerman et al. [96] for conducting design science research in the area of Information Systems (IS). This methodology is an integration of previous works in this area such as the framework defined by Hevner et al. [59] and the process model defined by Peffers et al. [97]. The result of this integration is a process model structured in three main phases, which can be divided into steps. How these phases are applied to this work is further described below and shown in Fig. 1.4-1.

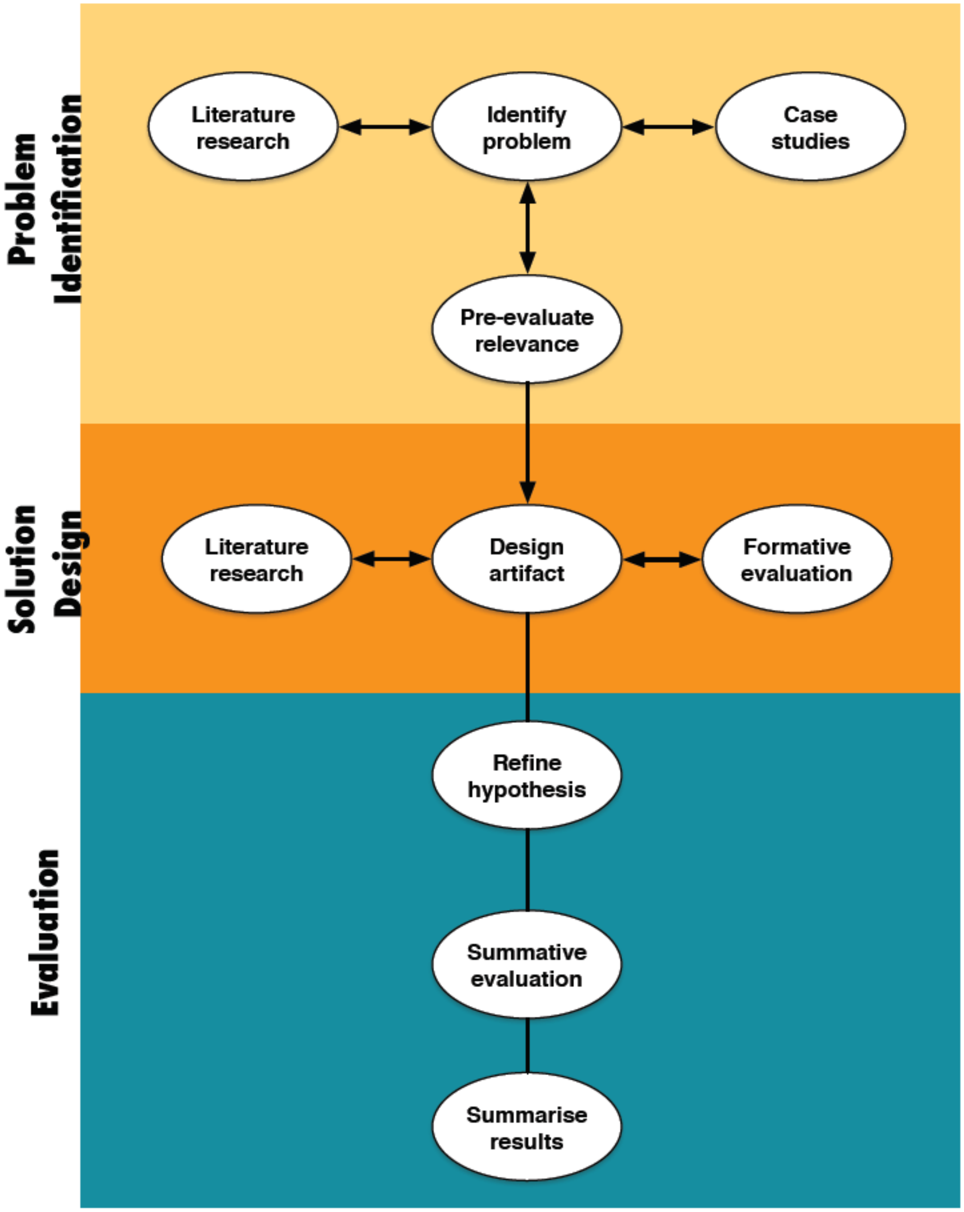


Fig. 1.4-1 Research methodology followed in this research work (adapted from Offerman et al. [96])

Phase 1. Problem Identification. This phase should address relevant research problems. It is divided into two steps, including *the identification of the problem* and *the pre-evaluation of its relevance.* To be relevant, research must address the problems faced and the opportunities afforded by the interaction of people, organizations, and information technology [59]. With the purpose of identifying a relevant problem, literature research and expert interviews or case studies can be used.

In the context of this work, this stage involved both *a literature review* and two *case studies*. The performance of this review was based on the systematic literature review process proposed by Kitchenham [69] in the area of software engineering. This review was conducted to characterize both key factors affecting design challenges in alarm visualization design and existing design knowledge reuse approaches that guide the design of alarm visualizations. The two case studies framed within the context of two projects, Energos and Emercien, allowed illustrating how such design factors operationalizes in practice. The purpose of Energos was to develop information and communication technologies that support new operational and organizational challenges caused by the Smart Grid. The goal of Emercien was to deploy sociotechnical platforms that promote effective and reliable collaboration in emergency management among different stakeholders, including first responders, decision and policy makers, volunteers and citizens. Once a relevant problem is identified, a pre-evaluation on the relevance has to be conducted at this phase. This pre-evaluation firstly includes creating a general research hypothesis, postulating a link between the problem space and the solution space. Second, this hypothesis should be evaluated by asking several practitioners if they agree with the hypothesis. The general research hypothesis formed during this phase was: *“If appropriate reusable items of design knowledge for alarm visualization design are used, it is possible to compensate the lack of design experience”.* In this work, a pre-evaluation wasn’t necessary due to the numerous examples of design knowledge reuse across different application domains.

Phase 2. Solution Design. This phase involves the proposal of a solution to be evaluated. It is divided into two steps, including supporting *literature research* and *artifact design.* To propose a solution, existing solutions and state-of-the-art have to be taken into account to ensure research rigour [59].

In this work, literature research involved a review of the existing types of approaches for reusing design knowledge. This review resulted on the selection of a *design pattern language* as a fitting approach for addressing and representing reusable alarm visualization design knowledge. Existing pattern languages for other purposes such as the design of visualization-based computational tools for complex cognitive activities [115], object-oriented software [51], user interfaces [136], interaction design [137], security systems [56], and hypermedia systems [88] were also taken into account. In order to define the design pattern language, design decisions were based on both an extensive literature review from different areas, including Alarm Management, Human Factors, and Visualization Design and a formative evaluation with expert designers. The purpose of this evaluation was twofold. On the one hand, identifying misunderstandings, or ambiguous terminology in order to refine the design pattern language. On the other hand, validating the descriptions provided by the design patterns according to the agreement of expert designers over its quality.

Stage 3. Evaluation. This phase should include both the demonstration of the use of the created artifacts to solve the research problem and the observation and measurement of how well they support the solution to this problem. With these purposes, this phase is divided into *the refinement of the prior research hypothesis* and *conducting* the *evaluation* of the created artifacts.

In accordance to Venable et al. [138], created artifacts in the field of design science research can be evaluated in terms of two specific dimensions such as the *quality* and *utility*. Accordingly, the prior hypothesis was split into smaller hypotheses that were simpler to evaluate. The following hypotheses were developed: (1) *"The design pattern language is well defined";* and (2) "*This design pattern language is useful for non-experienced designers".* To support the first hypothesis, diverse methods were utilized:

- (1.a) an *expert-based evaluation.* This evaluation sought to obtain feedback from expert designers regarding both the terminology and quality of the design patterns.
- (1.b) an *analytical evaluation*. This evaluation performed a prospective review of the context of the design pattern language to determine its quality.

To support the second hypothesis, two different evaluation methods were used:

- (2.a) a *descriptive evaluation* to validate the capability of the design pattern language of being used to create alarm visualizations in two different application domains.
- (2.b) an *experimental evaluation* with two iterations in order to evaluate the usability and efficacy of the design pattern language to allow non-experienced designers to reuse previous design knowledge.

1.5 Research Contributions

The research contributions of this work can be induced from the research questions previously formulated (see Section 1.3 Research Questions). Accordingly, for each research question, Table 1.5-1 shows its corresponding contributions.

Research Question	Research Contribution
RQ 1*. What is the design space for alarm visualization design?*	✓ Characterization of the design space for alarm visualization design in terms of recurrent design challenges and factors affecting them
RQ 2. *How to support non-experienced designers in the process of reuse previous design knowledge for designing alarm visualizations?*	✓ Systematization of reusable design knowledge on the design space for alarm visualization design ✓ A design artifact that non-experienced designers can use to reuse previous design knowledge to design alarm visualizations

Table 1.5-1 Summary table of the research contributions of this research work

1.6 Thesis Outline

This research work is structured in **six chapters**, four annex sections, and a references section. Excluding this chapter, in what follows, the contents of each chapter are briefly described:

Chapter 2. Background Research. This chapter presents the theoretical foundations of this work. It presents a review of both diverse descriptive models and design material for alarm visualization design in order to provide descriptions of the factors to consider when designing visualizations. It also reviews the notion of design knowledge reuse across domains in order to provide a description of why is needed in design.

Chapter 3. Problem Identification. This chapter describes the research problem undertaken by this work. Based on the previous literature reivew, it firstly frames the design space for alarm visualization design. Then, it characterizes the limitations of existing design knowledge reuse approaches to assist non-experienced designers when designing alarm visualizations.

***Chapter 4. A Design Pattern Language for Alarm Visualization Design*.** This chapter describes the proposed solution to the problem identified. The proposed solution is a design pattern language that captures, organizes, and combines reusable design knowledge for designing alarm visualizations in a more cohesive way to assist non-experienced designers.

Chapter 5. Evaluation. This chapter describes the evaluation process of the proposed solution. This evaluation assesses the quality and utility of the proposed design pattern language.

Chapter 6. Conclusions. This chapter presents the conclusions drawn from this work and discusses other application areas of the propose design pattern language. It also provides an outlook on future research and desribes some limitations of this research work.

Chapter 2. Background Research

Our Age of Anxiety is, in great part, the result of trying to do today's jobs with yesterday tools

- ***Marshall McLuhan,* Canadian Philosopher**

Adapting the previous definition of alarm visualization, in this research work alarm visualization design is defined as:

Definition 1. Alarm visualization design. *"The process of creating the visual representations by which alarm coding and messages are presented to control room operators".*

This design process requires designers to face multiple times two fundamental design challenges, including *how to display large volumes of alarms with multiple attributes such as typology, priority or location* and *how to assist human operators in the process of making sense of these large volumes of alarm information in order to get to know the state of the process being controlled*. These two design challenges involves understanding different factors affecting them related to specialized knowledge areas such as Alarm Management, Human Factors, and Visualization Design. However, no single designer can be an expert in every relevant area, and becoming proficient may require years of experience [51]. One relevant approach to assist designers in such design process is to reuse prior design knowledge instead of starting from *scratch*.

This chapter firstly presents a review of diverse descriptive models within the areas of Alarm Management, Human Factors, and Visualization Design in order to provide descriptions of the factors to consider when designing visualizations. Likewise, it examines the state of the art of design material that supports designers in the design of alarm visualizations. Secondly, this chapter discusses the notion of *design knowledge* *reuse.* A description of what is and why is needed in design is provided. It then leads to a discussion about its value within the context of visualization design.

2.1 Alarm Management

ANSI/ISA 8.2-2009 [64], one of the de facto standards for alarm management established by the International Society of Automation (ISA), defines an alarm as *"an announcement to the operator initiated by a process variable passing a defined limit as it approaches an undesirable or unsafe value"*. According to this definition, alarms are regarded as significant attractors of attention for human operators. Based on this idea, alarm management is primarily referred to as "*the process of understanding, designing, implementing, and operating an effective alerting capability for human operators"* [14]. In this way, the role of the human operator handling alarms may be firstly examined to determine how to design alarm visualizations. To this purpose, the next section reviews different models of human operators in response to alarms. Afterwards, a review of existing design rules in this context is provided. As a design rule is considered "*a rule that a designer can follow to provide direction for the design process"* [4].

2.1.1 Models of Alarm-Initiated Activities (AIA)

The notion of alarm-initiated activities refers to as *"the ensuing behaviours triggered by the presence of alarms"* [125]. Based on this notion, a variety of descriptive models have been proposed. These models may be split into two types: those models that don't distinguish operators' activities among operation situations and those that do.

Under the first distinction, most authors have agreed about characterizing alarm-initiated activities as a process comprising three-level activity stages. For example, Lees' model [79] comprises: (i) *detection* (detecting the fault); (ii) *diagnosis* (identifying the cause of the fault); and (iii) *correction* (dealing with the fault). Similarly, Rouse's model [108] establishes: (i) *detection* (the process of deciding that an event has occurred); (ii) *diagnosis* (the process of identifying the cause of an event); and (iii) *compensation* (the process of sustaining system operation). These models thus emphasize the role of the human operator as a problem solver when faults arise. However, they describe such role as static across different operation situations, without considering the diverse range of controlled process states and different operation goals. It means that these models conceive that human operators react in the same way and require the same pieces of information across operation situations. Nevertheless, in real work settings, human operators are dependent on the information provided by the alarm visualization, which can vary in number and priority according to the status of the controlled process. Alarm information has different priorities according to the severity of consequences that could be prevented by taking corrective action. Moreover, there exist different operation goals according to such status. According to Mattiason in his study on alarm systems from the operator's perspective [85], during normal operation the goal is to optimize, pushing towards constraints with a minimum of product quality giveaway. When a minor upset occurs, human operators' job is to bring the process back to normal operation. During a major upset she/he is expected to bring the process to the nearest safe state, and if disaster threatens, shut it down, and try to limit the consequences.

Under the second distinction, Stanton [125] has proposed the most complete and well-known descriptive model of alarm-initiated activities. In contrast to the previous models, this model distinguishes alarm-initiated activities between two operating situations: *routine events* involving alarms and *critical events* involving alarms. During critical events, even minor disturbances, the alarm systems will generate a huge, unmanageable amount of alarms [85]. For routine events, as shown in Fig. 2.1-1, this model is composed by a sequence of six generic activity stages: (i) *observe* (initial detection of abnormal conditions); (ii) *accept* (the acceptance of an alarm or receipt); (iii) *analyse* (the assessment of the alarm within the context of the task that is to be performed and the dynamics of the system); (iv) *correct* (to adjust so as to meet the required conditions of the system); (v) *monitor* (the assessment of the outcome of one's actions); and (vi) reset (to restore to normal operating conditions). For critical events, this model adds one activity stage to this sequence; *investigate*, which is depicted as *"seeking to discover the underlying cause of the alarm with the intention of dealing with the fault".* This model therefore not only establishes the distinction of alarm-initiated activities among operating situations but also emphasizes the need of an investigation process to diagnose the fault during critical events. During critical events, the cause of the failure can be not so clear. For this reason, it can be necessary to explore through different types and volumes of information in order to diagnose its cause. Moreover, human operators should be able to distinguish among several information sources such as direct telemetry from large-scale, distributed control systems or environmental data covering factors such as weather status or lightning strikes.

Fig. 2.1-1 Model of alarm-initiated activities proposed by Stanton [125]

From the point of view of alarm visualization design, this distinction of alarm-initiated activities between operation situations involves the necessity of presenting the information in a manner that always aids to the human operator [125]. Firstly, it is required to design alarm visualizations that effectively make a distinction of volumes of alarms among operation situations, in terms of either the number or the dimension of alarm information. Secondly, it is needed to present the alarm information in a form that matches with different operation goals.

2.1.2 Designing for Alarm-Initiated Activities (AIA)

The critical nature of alarm visualizations is highlighted by the fact that a high proportion of incidents are caused by operators' errors [55]. Several industrial manufacturers, and vendors from worldwide have defined a set of design guidelines and standards that document what are considered the expectations and best practices for designing effective alarm visualizations. Similar understandings, philosophical underpinnings, and end effects drive these works.

For the first type of design rule, the Abnormal Situation Management (ASM) Consortium defines a collection of guidelines for effective operator display design [16]. This consortium is a group of leading companies and universities involved with process industries that have jointly invested in research and development to create knowledge, tools and products designed to prevent, detect, and mitigate abnormal situations that affect process safety in operation environments. These guidelines are neither a standard nor a regulation. They just promote best practices that facilitate the design of alarm visualizations based on several studies conducted with operators. Similarly, EEMUA 191 [40] can be characterized as a design guide. However, it has served as the de facto standard for designing alarm systems until 2009, with the appearance of the ANSI/ISA 18.2-2009 [64]. It mainly raises a number of alternatives from which practitioners must select the most appropriate for their process. The Engineering Equipment and Materials Users' Association (EEMUA), based in the United Kingdom, produce it. This association develops standards for organizations in the United Kingdom that operates process and power plants, utilities and other industrial facilities.

For the second type of design rule, more recently, the International Society of Automation (ISA) has defined a comprehensive standard to improve safety, the ANSI/ISA 18.2-2009 [64], built on the recommendations of EEMUA 19. The key difference with the previous works is that the ANSI/ISA 18.2-2009 is a standard, not a guideline or a recommended practice, and it was developed in accordance with stringent ANSI methodologies. As such, it is regarded as a recognized and generally accepted good engineering practice by regulatory agencies. However, it is in the process of being adopted as an international standard.

As a result of the review of these design rules, it is possible to induce a list of key points in common for guiding the design of alarm visualizations. These key points are listed in what follows:

- **Human factors and limitations.** The design of alarm visualizations should ensure that the alarm system remains usable in all process conditions, by ensuring that unacceptable demands are not placed on operators by exceeding their perceptual and cognitive capabilites.

- **Multiple visual displays formats.** To support the different functions of an alarm system, multiple visual display formats may be required. It refers to a combination of separate displays such as alarm tiles and integrated displays such as alarms integrated into process displays.
- **Display hierarchy.** Alarm visualizations should provide different levels of details and be arranged to allow the operator to *drill-down* to increased levels.
- **Navigation**. The navigation through these visualizations should facilitate quick, direct access to primary visualizations and minimal keystrokes to secondary and associated visualizations.
- **Window management.** It is recommended to define a limit for the maximum number of simultaneous overlap windows.
- **Priority information**. Alarm information should be presented in a priority layer where any changes are brought immediately to the operator's attention.
- **Colour coding.** Use a minimum of colour codes across display hierarchy levels. Consistent, distinguishable colour codes allow operators to learn the codes and the meaning behind them.
- **Embedded trends**. It is recommended to display embedded trends in order to draw attention to abnormalities and deviations. Alarm visualizations should make a distinction between changing values that take exact readings and general alarm trends.

In summary, what these rules highly recommend to designers is the achievement of a balance between the need for content and simplicity of presentation. This balance must result in intuitive and flexible presentation means to accessing the required alarm information by human operators. However, they describe just the minimum acceptable for achieving this balance. By design, they focus on *what to do* rather than *how to do it.* It is designers' task therefore to establish how these rules are applied to the design of alarm visualizations. Moreover, it is also designers' task to ascertain how these rules should be combined in order to generate a complete alarm visualization design.

2.2 Human Factors

In consistence with the previous characterization of alarm-initiated activities, the human factors and ergonomics field recognizes the need of designing alarm visualizations with all the human operator's capabilities, goals and needs in mind [107][44]. In handling alarm information, Situation Awareness (SA) can be characterized as the most relevant goal for human operators. In particular, SA can be defined informally as *"being aware of what is happening around you and understanding what that information means to you now and in the future"* [260]. However, building and maintaining this SA can be a difficult process when people supervise complex and dynamic processes. As an example, in real-time operations, the analyses of recent operating problems have shown that the ability of human operators for acquiring SA is one of the major factors that affects the propagation of failures [55].

With the purpose of designing effective alarm visualizations, it is firstly required to look more closely at the meaning of this concept. There is a large body of literature in SA, and this continues to be an active area of research. In particular, reviews of definitions and theories from varied sources [109][55] provide a clear indication of the variety of approaches about SA. Thus, the focus of the next section is on to provide a review of these SA approaches in order to characterize those to be principally considered for designing alarm visualizations. Afterwards, a review of the most applied design rules for designing for SA is conferred.

2.2.1 Models of Situation Awareness (SA)

As Rousseau, Tremblay and Breton [109] firstly observed, there exists a generally accepted duality of SA as a *state* or a *process*. Under the first conceptualization, Micah Endsley has provided the most highly recognized descriptive model of SA. Endsley's well-known SA definition [44] describes SA as *"the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning and the projection of their status in the near future".* Thus, Endsley uses the term to define a state of knowledge and she describes the associated process as situation assessment. In particular, Endsley's model (see Fig. 2.2-1) includes two parts: a core SA model, and a set of factors affecting SA. The core SA model is the basis for much of the current thinking about SA. It is a three level model comprising *Level 1, perception of the elements in the environment*; *Level 2, comprehension of the current situation*; and *Level 3,* *projection of future states*. In what follows, a brief description of the relevant aspects of each level is provided.

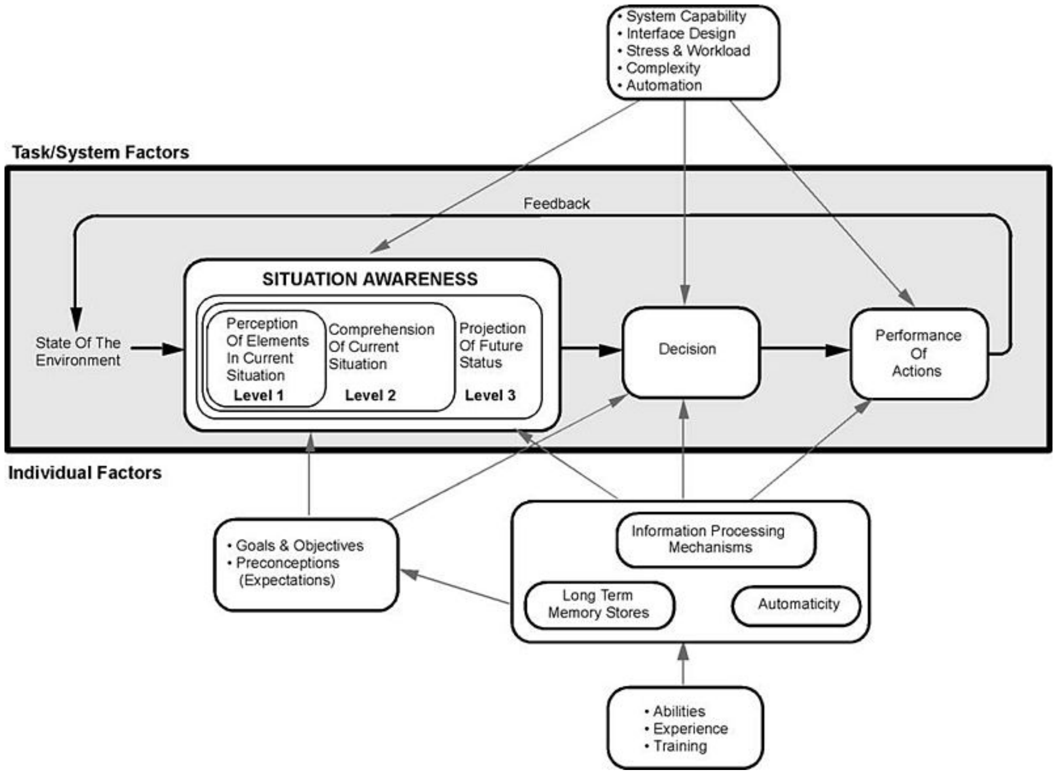


Fig. 2.2-1 Model of situation awareness in dynamic decision-making [44]

The Perception level. This level is the first step in achieving SA. It is referred to perceive the status, attributes, and dynamics of the relevant information elements in the environment. Perception may come through visual, auditory, tactile, taste, olfactory senses, or a combination. In cognitive terms, it involves interaction with long-term memory (comparing the information elements with what is already known); it is under attentional selection (modulated by the operator's selective attention processes as determined by task requirements); and the information content is held in active working memory.

The Comprehension level. The second step in achieving SA is defined as understanding what the data and cues perceived mean in relation to relevant goals and objectives. Consequently, comprehension is based on a synthesis of disjointed Level 1 elements, and a comparison of that information to one's goals. In cognitive terms, mental models stored in long-term memory provide a basis for Level 2 SA.

The Projection level. Once the person knows what the information elements are and what they mean in relation to the current goal, this level is described as the ability to predict about the states of the environment in the near future. A person can only achieve Level 3 SA by having a good understanding of the situation (Level 2 SA) and the functioning and dynamics of the system they are working with. This projection allows people to be proactive in making decisions, avoiding many undesirable situations, and also very fast to respond when various events do occur.

In this view of SA, thus, information elements are processed to yield meaning. Accordingly, Endsley suggests that a common solution to human information processing limitations is to design methods to facilitate processing of more information through limited processing channels.

Under the second conceptualization of SA, several authors have agreed about considering SA as a label for a range of cognitive processes or processing activities. For example, Dekker and Lutzhoft [33] describe the concept of SA as *"an intrinsic feature of the functional relationship between the environment and the person"*. This empiricist view of SA breaks down the process into perception of information elements and is highly consistent with current ideas about *sense making* as an active strategy for dealing with a complex world. Klein et al. [70] define the *sense-making* concept as *"the ability or attempt to make sense of an ambiguous situation. It is the process of creating situational awareness and understanding to support decision-making under uncertainty. It is an effort to understand connections among people, places, and events in order to anticipate their trajectories and act effectively".* In this conceptualization of sense making, Klein et al. establish that when people try to make sense of events, they begin with some perspective, viewpoint or framework, which he calls frame. This frame can include stories, maps, organizational diagrams, or scripts, and it can be used in subsequent and parallel processes. In particular, this frame defines what count as data, as well as shape the data. Furthermore, the frame changes as people acquire data. Accordingly, he defines the Data/Frame theory of sense making (see Fig. 2.2-2) that posits a closed-loop transition sequence between: (i) *mental model formation* (which is backward looking and explanatory), and (ii) *mental simulation* (which is forward looking and anticipatory).

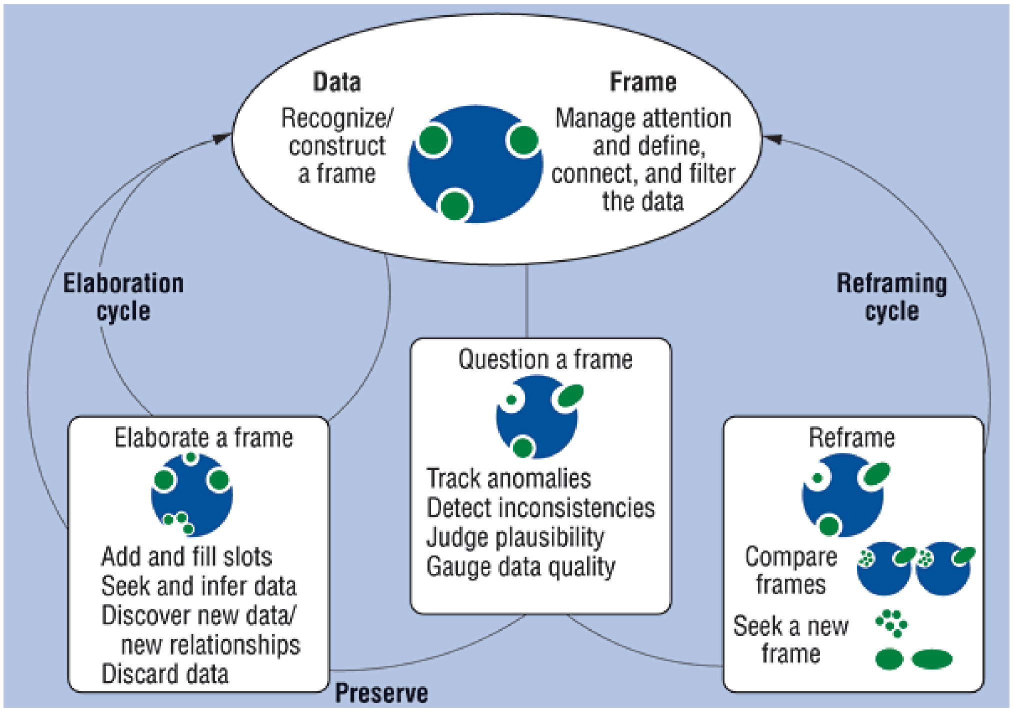


Fig. 2.2-2 The Data/Frame theory of sense making [71]

Therefore, while the sense-making perspective on SA may acknowledge the existence of elements, its focus is on the role they play constructing a plausible story of what is going on, not for building an accurate mental model of an external world. Dekker and Lutzhoft also observe that few theories of SA acknowledge this, instead directing their attention to the creation of meaning from information elements and the future projection of that meaning. Thus, in studying SA, rather than examining the lack of correspondence between actual and experienced worlds, they recommend examining the decisions makers unfolding experience of the situation in which they found themselves. This approach is related to the study on how problem solving is achieved in complex and uncertain real world situations, which has become known as Naturalistic Decision Making (NDM). NDM contrasts with previous theories and establishes that features of the situation and the decision maker such as the level of expertise dictate the form of decision-making processes adopted, thus, understanding the context surrounding the decision process is essential [80].

More recently, Guttromson et al. [55] have extended this SA duality to two new different points of view. In particular, they consider that there is still not a clean separation between the conceptualizations of SA as a state versus a process. As an evidence of that, they point out Klein's work, which indicates that situation assessment is a very broad concept that covers situation awareness content (as defined by Endsley) as well as actions (process). As a possible source of confusion, they suggest that these different points of view reflect two different basic approaches to the question of SA. One point of view is *operator-focused* and it concerns the properties and mechanisms of the operator as they determine SA. The other point of view, is *situation-focused*, holds that SA is determined by the environment or situation in which the operator works. In what follows, these different points of view are briefly reviewed.

The operator-focused approach is concerned with the set of cognitive processes supporting the production of mental representations corresponding to the SA state. Such processes are by definition a property of the human operator. This approach follows an information-processing framework that considers a mental representation of the world to be based on processing with specific functions such as perception, comprehension, and projection (as defined by Endsley); or information extraction, information integration, mental picture formation, and projection or anticipation. However, most of the process-based definitions do not follow generally accepted human information processing models that identify a network of processes like perception, attention, and memory.

The situation-focused approach is concerned with mapping relevant information in the situation onto a mental representation of that information within the operator. State-oriented definitions limit the description of processes involved in SA, in line with Gibson's fundamental concept of direct perception [52] that involve such principles as: (i) all information necessary for perception is contained in the environment; and (ii) perception is immediate and spontaneous. This implies that in order to understand perception, the priority must be on understanding the environment (independent of underlying processes). The situation-focused approach provides a basis on which to define SA as a state, defining a situation in terms of events, objects, systems, other persons, and their mutual interactions. This is domain-dependent. Thus, it is much more than awareness of the distribution in space of objects within a contextual environment. It has to include task goals, criteria of performance, and cues in the environment.

The above review shows thus the complexities and divergent concepts that can be encompassed by the concept of SA in general. However, delving more specifically into SA while designing for supervision of complex and dynamic processes reveals that the sense-making perspective of SA has been introduced as a process that should be widely considered [55]. In contrast to the traditional view of SA proposed by Endsley, this perspective describes the process of how SA is achieved when the available information is uncertain or conflicting, and maintained or recovered after surprising events, which are typical situations that a human operator has to deal with. Furthermore, according to Flach, Mulder, and van Paasen [47], a poor SA in this context reflects the lack of a basis for decomposing complex information or data into coherent chunks. Therefore, it is suggested that supporting SA in this context is not about providing more information, but rather about providing priorities and clarity about preferences to help the operator understand what matters.

2.2.2 Designing for Situation Awareness (SA)

Rather than displaying information that is centred on the sensors and technologies that produce it, computer-based information displays such as alarm visualizations should present this information in ways that fit the goals, activities, and needs of the operators [44]. Aiming at addressing these issues, Endsley has defined the most complete and applied design rules for designing for SA [45][67]. In particular, she defines fifty design principles. These principles are based on a model of human cognition involving dynamic switching between goal-driven and data-driven processing, and feature support for limited operator resources. These design principles underpin not only computer-based information display design issues, but also how to design automated systems, dealing with complexity or uncertainty. However, the focus of this section is on those eight principles about displaying information:

- **Organize information around goals.** Information should be organized in terms of the operator's major goals, rather than presenting it in a way that is technology-oriented (displayed based on the sensors or systems which created the information).
- **Present Level 2 information directly.** As attention and working memory are limited, the degree to which displays provide information that is processed and integrated in terms of Level 2 SA requirements will positively impact.
- **Provide assistance for Level 3 SA projections.** One of the most difficult parts of SA is the projection of future states of the system. Projection requires a fairly well developed mental model. Therefore, displays that

allow operators to anticipate possible occurrences should be provided. They should provide sufficient resolution in time to ensure that rapidly changing variables can be observed and interpreted.

- **Support global SA.** Global SA is defined as *"a high level overview of the situation across operator goals".* Consequently, displays that provide the operator with global SA should be provided.

- **Support trade-offs between goal-driven and data-driven processing.** Designs need to take into consideration both top-down and bottom-up processing. The design of the system around operator goals (Principle 1) will support goal-directed processing. The big picture display that supports global SA (Principle 4) will support data-driven processing by directed the operator as to where to focus attention to achieve high priority goals. The key is to ensure that these two approaches complement each other.

- **Make critical cues for schema activation salient.** In that mental models and schemata are hypothesized to be key features used for achieving the higher levels of SA in complex systems, the critical cues used for activating these mechanisms need to be determined and made salient in the interface design.

- **Take advantage of parallel processing capabilities.** The ability to share attention between multiple tasks and sources of information is important in any complex system. System designs that support parallel processing of information by the operator should directly benefit SA. While people can only visually take in so much information at one time, they are more able to process visual information and auditory information simultaneously.

- **Use information filtering carefully.** Presenting information in a clear and easy to process manner, with the operators in charge of determining what they will look at when, is far better than computer-driven strategies for providing only subsets of information.

These design principles are based on the Endsley's conceptualization of SA as a knowledge state. They provide therefore guidance to the design of alarm visualizations that allow achieving a knowledge state, which can be referred to as either knowledge of current data elements, or inferences drawn from these data, or predictions that can be made using these inferences. However, they leave aside the support to the process of achieving these kinds of outcomes, the strategies and the barriers encountered, which have been conceptualized as the sense-making process. As it was characterized before, this sense-making process should be considered while designing alarm visualizations for control systems. Similarly, as the design rules from the *Alarm Management* field, these principles are too abstract and their specific implementation will depend on the circumstance and the designer's creativity.

2.3 Visualization Design

The consideration of the sense-making perspective of SA while designing alarm visualizations is consistent with current ideas about sense making as the main purpose of visualization. As a result, visualization can be portrayed as a suitable tool to support the operator's SA while designing alarm visualizations. According to Card et al. [20], until recently, the term visualization meant "*constructing a visual image in the mind".* However, the most common understanding of visualization has changed over time and is now mostly referred as *"the use of computer-supported, interactive visual representations of data to amplify cognition".* Thus, from being an internal construct of the mind, a visualization has become an external artifact to amplify *cognition* [141]; understanding cognition as the mental processes of knowing, including awareness, perception, reasoning, and judgment. In particular, visualization has three major goals: (i) *Presentation.* It refers to an efficient and effective communication of facts that are fixed a priori; (ii) *Confirmatory analysis.* It can be described as a goal-oriented examination of existing hypotheses with the aim of confirming or rejecting them; and (iii) *Exploratory analysis.* It is a typically undirected search for new information like structures and trends without initial hypothesis. However, with the rise of computers, the movement from static images to interactive visualizations has turned exploratory analysis into the most relevant goal of visualization. Moreover, according to Munzner [90], visualization should be especially applied when the goal is to augment human capabilities in situations where the problem is not sufficiently well defined for a computer to handle algorithmically. In the case of alarm-initiated activities, it requires human judgment to make the best possible evaluation of overwhelming amounts of incomplete, inconsistent, and potentially deceptive alarm information in the face of rapidly changing situations. During these situations, visualization can provide an ability to comprehend huge amounts of information and allows the perception of emergent properties that were not anticipated [141].

Similarly, it can facilitate understanding of both large-scale and small-scale features of the information [132]. These capabilities are related to the concept of *visual scalability*, which is defined as *"the capability of visualizations to effectively display large data sets, in terms of either the number or the dimension of individual data elements"* [41]. To this purpose, diverse relevant factors including human perception, display size, and visual metaphors need to be considered.

The next section firstly reviews descriptive models of visualization. These descriptive models provide descriptions of the elements that compose a visualization. Secondly, the visualization design process is reviewed. It describes the steps that designers use in creating visualizations. Finally, relevant design rules are depicted. These design rules can be used to provide direction for the visualization design process.

2.3.1 Visualization Models

Despite their variability, visualizations can be systematically analysed. In particular, as Card et al. [20] states, a visualization can be described as *"the mapping of data to visual form that supports human interaction in a workplace for visual sense making".* In accordance to that, different models of visualization have been proposed, which have agreed about characterizing a visualization as consisting of a set of mappings, combined in a number of feedback loops.

According to the model proposed by Ware [141], a visualization includes four basic stages (see Fig. 2.3-1). The four stages consist of: (i) *the collection and storage of data*; (ii) *a pre-processing stage designed to transform the data into something that is easier to manipulate*; (iii) *mapping from the selected data to a visual representation, which accomplished through computer algorithms that produce an image on the screen*; and (iv) *the human perceptual and cognitive system.* The longest feedback loop in this model involves gathering data. A user may choose to gather more data to follow up on an interesting lead. Both the physical environment and the social environment are involved in this loop. Another loop controls the computational pre-processing that takes place prior to visualization.

Fig. 2.3-1 Model of visualization proposed by Ware [141]

Similarly, Card et al. [20] has proposed the most well known model of visualization, which is, in fact, referred to as “the visualization reference model” (see Fig. 2.3-2). This model defines three main stages: (i) *data transformations* (raw data, which is defined as data in some idiosyncratic form, is transformed into data tables, which are described as canonical descriptions of data in a variables x cases format extended to include metadata); (ii) *visual mappings* (data tables are transformed to visual structures, which are defined as structures that combine values and available vocabulary of visual elements); (iii) *view transformations* (visual structures can be further transformed by view transformations, until it finally forms a view that can be perceived by human users). In this model, there is a flow back from the human into the transformation themselves, indicating the adjustment of these transformations by user-operated controls.

Fig. 2.3-2 The visualization reference model proposed by Card et al. [20]

Both models highlight thus that visualization is not simply about presenting information. Rather, it is more of a dialogue between the user and the data, where the visual representation is simply the interface or view into the data. In this dialogue, the user observes the current data representation, interprets and makes sense of what he or she sees, and then thinks of the next question to ask, essentially formulating a strategy for how to proceed [124][20]. Accordingly, visualizations should be designed to support this human-information discourse. To achieve that, the data types and tasks performed by the user, as well as his physical and the social environment should be considered by the designer [132].

2.3.2 Design Process

Visualization design can be defined as *“the process of designing information to match the processing characteristics of the human visual system”* [147]. With this mission, this process carries out the previous mappings through two main approaches that are different yet complementary: the *data-oriented design approach* and the *human-centred design approach.*

Under the data-oriented design approach, the accomplishment of these mappings is mainly driven by data attributed characteristics. For this reason, most previous relevant work has been focused on guiding the visualization design by providing taxonomies of visualization techniques using a data-centric point of view. As an example, Card and Mackinlay [19] started constructing a data-oriented taxonomy, which was subsequently expanded in [20]. This taxonomy divides the field of visualization into several subcategories: *Scientific Visualization*, *Geographical Information Systems* (GIS), *Multi-dimensional Plots*, *Multi-dimensional Tables*, *Information Landscapes*, and *Spaces*, *Node and Link*, *Trees* and *Text Transforms*. Similarly, Shneiderman [120] defined the well-known task-by-data type taxonomy according to eight visual data types: *temporal*, 1-dimensional (1D), *2-dimensional* (2D), *3-dimensional* (3D), *multi-dimensional* (multiD), *tree*, *network*, and *workspace*. These taxonomies thus emphasize the relevance of the type of data that it is needed to look at in the visualization design. However, they describe such relevance as static across different tasks and environments, without considering the specific needs and limitations of some intended users.

Under the human-centred design approach, these mappings are not only driven by data characteristics but also by the user’s tasks and the environment. Over the last decade, applying this approach has led to the definition of the concept of *human-centred visualization design,* the general idea of which is that visualization design should take into account user's needs, skills, and limitations, due to its own definition. Based on this ground, and trying to provide some guidance to visualization designers, several authors such as Munzner [91], Wassink et al. [142], Kerren et al. [68] or Zhang et al. [147] have proposed models and design frameworks that describe the structure of tasks, users, and functions. Although these models have clear differences among them, they concur in highlighting the characterization of the user's tasks at different levels of abstractions as the decomposition factor required for making visual encoding and interaction decisions in visualization design.

2.3.3 Design Rules for Visualization Design

According to Ware [141], previously to create effective computer-mediated visualizations, it is required to understand how they enable cognition as a basis for design decisions. The subjects related with aids to cognition are one of the main focuses of cognitive science. Accordingly, cognitive scientists have studied visual representations and the larger class of external aids to cognition. As a result of these studies, some basic design principles for developing effective depictions have been defined. As an example, Norman [94] defines three basic principles: (i) *Appropriateness Principle - The visual representation should provide neither more nor less information than that needed for the task at hand. Additional information may be distracting and makes the task more difficult; (ii) Naturalness Principle - Experiential cognition is most effective when the properties of the visual representation most closely match the information being represented;* and (iii) *Matching Principle - Representations of information are most effective when they match the task to be performed by the user. Effective visual representations should present affordances suggestive of the appropriate action.* Another prominent cognitive scientist, Tversky et al. [135] has suggested the following two basic principles: (i) *Principle of Congruence - The structure and content of a visualization should correspond to the structure and content of the desired mental representation. In other words, the visual representation should represent the important concepts in the domain of interest*; and (ii) *Principle of Apprehension - The structure and content of a visualization should be readily and accurately perceived and comprehended.* These principles firstly underlie the importance of research in perception. In accordance to that, Gestalt theories [131] have been widely applied (see Fig. 2.3-3).

Fig. 2.3-3 Application examples of Gestalt theories [131]

These theories define the rules according to which human perception tends to organize visual elements into a unified whole, also referred to as groups. Secondly, these principles underscore that the biggest challenge in choosing a visual representation is to find the right one, not just any one, for the task at hand. Consequently, the next step to take in developing effective visualizations has been established as the formal definition of the different types of visualizations [132].

With the purpose of addressing this challenge, the cartographer Bertin [10] has provided the most applied approach to communicate information by visual means. In this approach, mainly based on his own judgment, Bertin considers the space of possible visual representations as a visual language. The spatial and visual attributes of the representation encode the information using the rules of the language. In particular, he describes *marks* as the basic units of a visual representation. These marks can be: (i) *Points* (dimensionless locations on the plane, represented by signs that obviously need to have some size, shape or colour for visualization); (ii) *Lines* (they represent information with a certain length, but no area and therefore no width); (iii) *Areas* (they have a length and a width and therefore a two-dimensional size); (iv) *Surfaces* (they are areas in a three-dimensional space, but with no thickness; and (v) *Volumes* (they have a length, a width, and a depth. They are thus truly three-dimensional). Similarly, he also defines a given number of methods through which these units can be modified. These predefined modifications are called *visual variables*. In particular, Bertin defines seven visual variables (see Fig. 2.3-4) classified under two different categories: (1) *Planar variables* (defined as visual variables related to the x and y position on the map. *Position* is the only visual variable that belongs to this category); and (2) *Retinal variables* (defined as visual variables that can be processed automatically by the human eye. These retinal variables are highly related with the preattentive variables defined later by Ware [141]. In particular, these retinal variables are *orientation*, *shape*, *colour hue, texture, colour value,* and *size)*.

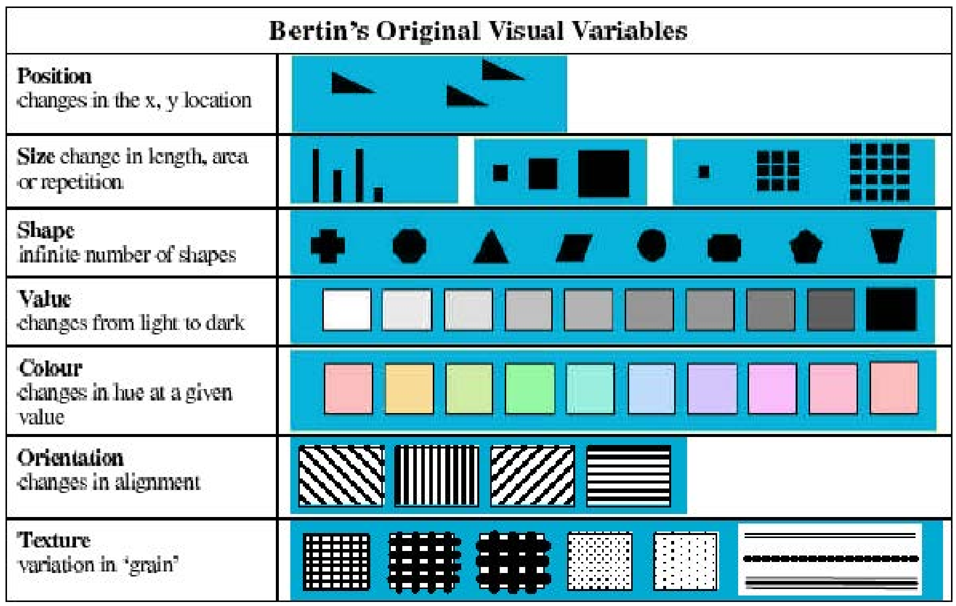


Fig. 2.3-4 Summary table of the seven visual variables proposed by Bertin [10]

Likewise, he defines different *visual variable characteristics*. These visual characteristics are: (i) *Selective*. A visual variable is said to be selective if a mark changed in this variable alone makes it easier to select that changed mark from all the other marks. This task is about the selection of an individual mark as distinct from other marks); (ii) *Associative*. A visual variable is said to be associative if marks that are like in other ways can be grouped according to a change in this visual variable. This means that several marks can be grouped across changes in other visual variables); (ii) *Quantitative*. A visual variable is said to be quantitative if the relationship between two marks differing in this visual variable can be seen as numerical. These are not necessarily precise numerical readings but are often read as ratios of one mark to another); (iv) *Ordered.* A visual variable is said to be ordered if changes in this visual variable support ordered readings. That is a change in an ordered visual variable will automatically be read as either more or less); (v) *Length*. Length is a slightly different kind of characteristic. The length of a visual variable is the number of changes that can be used and still retain in the task supporting characteristics that are usually associated with this visual variable. Thus, the choice of the most appropriate visual variable to represent each aspect of information depends on their characteristics. In particular, according to Bertin, shape and orientation variables are not selective. Conversely, only the visual variables size and colour value are said to have perceptual dissociative characteristics. With dissociative visual variables, it is easier to detect visual variations among the signs themselves, than to visually form groups of similar symbols across other visual variables. Furthermore, Bertin ranks these visual variables in an explicit sequence. In particular, he ranks the size as a higher order variable, which posses a greater number of perceptual characteristics. On the contrary, visual variables such as orientation are ranked as lower order variables that may only have associative characteristics. More recently, the visualization community has attempted to systematically apply this visual language to the automatic generation of visualizations. The most relevant example of that is the Automated Presentation Tool (APT) developed by Mackinlay [83]. This tool automatically designs representations based on Bertin's ideas. In particular, APT searches over a space of possible visual representations, evaluates them based on *expressiveness* and *effectiveness* criteria, and choses the best one. The expressiveness criterion states *"a visualization is expressive if it encodes all relevant information and only that information".* That means that the scientist may see all information he wants to examine. In like manner, the effectiveness criterion states *"a visualization is effective if it presents all the information clearly".* This excludes cluttered visualizations.

The above review shows thus the multidisciplinary nature of the visualization design, including design principles and visual languages from a number of fields such as cognitive science and cartography. These design principles and visual languages, respectively, allow designers to understand how visualization enables human cognition and identifying the basic units of a visualization. Nevertheless, most of them do not suggest how to apply them to define specific visualization designs. They assume an absolute validity across design situations while usually they lack describing them. A result of that are bad design choices or misinterpretations by designers.

2.4 Design Knowledge Reuse

Knowledge plays an important role in design and managing this knowledge is a concern [3]. All use of knowledge could be qualified "reuse" in the sense that knowledge is based on the processing of previously encountered data, experience, and representations constructed in the past [140]. However, trying to clarify the notion of knowledge, a number of commonly used classifications in the fields of design and engineering [24][84] can be distinguished: *tacit knowledge, implicit knowledge*, and *explicit knowledge*; *product knowledge* vs. *process knowledge;* and *compiled knowledge* vs. *dynamic knowledge*. In what follows, a brief description of each classification of knowledge is provided.

Tacit knowledge is that knowledge that resides in people's heads and may be destined to remain there. It is tied to experiences, intuition, unarticulated models or implicit rules of thumb. It is generally gained over a long period of time with learning and experience, is difficult to express, and can only be transferred by the willingness of people to share their experiences. On the contrary, explicit knowledge is that knowledge that has at minimum been captured and articulated and has ideally been codified, that is, documented, structured and disseminated. This knowledge creates the intellectual platform necessary to build and manufacture a design product. The laws of physics used for calculations are an example of explicit knowledge. A third category is implicit knowledge, which is not easily articulated by the person possessing it, but can be elicited and articulated by others. An example of implicit knowledge is the strategy adopted by an experienced designer to undertake a particular task in the design process.

Product knowledge includes various pieces of information and knowledge associated with the evolution of a product throughout its lifecycle. This includes requirements, various kinds of relationships between parts and assemblies, geometry, functions, behaviour, various constraints associated with products, and design rationale. On the other hand, process knowledge is concerned with the activity of designing itself. Process knowledge can be in turn classified into *design process knowledge*, *manufacturing process knowledge*, and *business process knowledge*. Design process knowledge, which can be encoded as methods in a product representation, provides mechanisms for realizing design details at various stages of the product lifecycle. Manufacturing process knowledge is mainly concerned with activities associated with the manufacturing floor. Finally, business process knowledge includes all processes associated with marketing, strategic planning, and other associated functions. While product and process knowledge are not independent of each other, they refer to distinct aspects of the knowledge dimension, and hence merit separate consideration.

Compiled knowledge is essentially knowledge gained from experience that can be compiled into rules, plans or scripts, cases of previously solved problems, etc. In compiled knowledge the solutions are explicit. Dynamic knowledge encodes knowledge that can be used to generate additional knowledge structures, not covered by compiled knowledge. In dynamic knowledge the solutions are implicit.

Bringing knowledge forward and making it explicit and formal is the principle output of the design process, which can be referred as *design knowledge.* Design knowledge is defined as "*the conceptual ideas, lessons, and representations captured in the design artifacts created or collected when solving a design problem"* [84]. This definition characterizes the design process as an activity that generates knowledge, and implicitly this knowledge can be reused. For instance, design solutions from prior problems can provide useful starting points for new problems, serve as references for comparing or explaining new ideas, and provide access to relevant design discussions [118]. Reuse can also improve design efficiency and lead to higher quality outcomes [49] [118]. However, how well the benefits of reuse can be realized depends on how well prior design knowledge can be stored, accessed, and retrieved. Much of this research has been conducted within software engineering, leading to a classification of design knowledge reuse approaches, divided into two broad categories [8]: *code reuse* and *knowledge reuse.*

Code reuse includes different approaches to organize actual code and incorporate it into software such as libraries of modules, code fragments, or classes and the use of off-the-shelf software. Code repositories can be considered design knowledge bases. However, though it is argued that the best mechanism to communicate design is the code itself, sharing design is not the same as sharing design knowledge. It is limited to programming and its complexity often encapsulates complex logic that would take years to develop from scratch. More recently, a distinction was made between the traditional view of code reuse and an emerging trend [103]. White-box reuse involves searching for code components, modifying them for use, and then depositing the component for others for use. This type of reuse involves the use of reuse repositories, an area for which many strategies and searching methods are researched. Black-box reuse, on the other hand, is a new trend that involves using components without modifications. Although black-box reuse requires that developers know the functionality of the component and how to interface with it, it reduces the need to search for and modify components.

Knowledge reuse refers to approaches to organizing and applying knowledge about software solutions, not to organizing the solutions themselves. It mostly distinguishes between three basic methods: *principles*, *processes*, and *patterns*. Principles are high-level concepts that guide the entire design process. For example, as it has been defined by Card et al. [20], *“visualizations should provide neither more nor less information than that needed for solving the problem”.* Processes are how designers put the principles into practice such as guidelines that explains how to conduct a focus group, how to run a survey, etc. Patterns are a language, a common vocabulary that allows designers to represent and articulate recurring design problems. For instance, the *Action Button* pattern [136], which has been widely applied to design user interfaces, solves a common problem about letting know the user what element on the interface can and cannot be clicked on. However, the most successful attempt to codify software design knowledge is the patterns approach. Originally proposed by Alexander et al. [6] for the design of buildings and towns, patterns are reusable components encapsulating design knowledge. They include information such as context of use, conflicting forces, and potential solutions. Patterns were adopted by the software engineers and developed a following over the years largely due to Gamma et al. [51], who proposed the reuse of patterns for software development.

Research on design knowledge reuse in HCI is has been also done, but the topic does not have the large following software engineering community has to date. The benefits of reuse on usability have been demonstrated. There are a number of authors that looked at the reuse of patterns for HCI. Borchers [12] argued for the use of design patterns to capture HCI knowledge. He mentioned the need for the encapsulation of the designers’ experiences, methods, and values into patterns. Some have created patterns for specific domains. For example, Landay and Borriello [78] created patterns for ubiquitous computing. Their goal was to apply them within a field by documenting lessons learned and passing them on to new designs. Such research efforts into the reuse of patterns provide impetus behind the argument to consider other forms of reusable design knowledge to benefit HCI. Claims are a form for recording knowledge proposed by Carroll and Kellog [21]. Claims were even included as a part of a design pattern structure by Hughes [61]. Similarly, Sutcliffe [128] proposed and spearheaded efforts to make claims into reusable design knowledge components, lasting well beyond the designs they were initially created for. An extensive structure for claims and the idea of storing claims in a library was presented [129][128].

The above review shows thus that there is great diversity of design knowledge reuse approaches across fields to promote the level of reusing the accumulated design knowledge. However, according to Krueger's dimensions for characterizing reuse approaches [76], it is possible to distinguish them in terms of its reusable components and the way these components are abstracted, selected, specialized, and integrated into the design process. In particular, abstraction is the essential feature in any reuse approach. Without abstractions, designers would be forced to sift through a collection of reusable components trying to figure out what each component did, when it could be reused, and how to reuse it. Similarly, selection is a required feature to guide designers as they search for components during a design process. After selecting a reusable component, the designer specializes it through parameters, transformations, constraints, or some other form of refinement. Finally, a designer combines the collection of selected and specialized components into a complete design artifact. In keeping with such dimensions, design patterns can be characterized as to be one of the most useful means of helping designers locate, compare, and select time-tested design solutions across fields.

2.5 Design Knowledge Reuse for Visualization Design

Visualization design can be defined as *"the process of defining a visual representation that can be processed by efficient human visual mechanisms"* [147]. With the growing emphasis on visualization as a mechanism for analysing and exploring large and complex data sets, visualization research recognizes the need to improve the efficiency and effectiveness of the visualization design process [25][132][26]. It acknowledges the need of reusing prior design knowledge versus the execution of a simply creative design process. Aiming at addressing this need, a number of design knowledge reuse approaches have been applied.

Some researchers as Mackinlay [83] have used formal graphical specifications, including visual languages and composition rules, to automate the construction of visualizations more efficiently and systematically. Similarly, Wilkinson [144] has proposed a language to construct statistical graphs using graph algebra. Although these approaches are very promising, they are too abstract. They depict the important characteristics of visualizations but they do not support the how, which must rely on designer's experience. Moreover, they have not been targeted at visual representations in support of exploratory processes such sense making. On the contrary, they are oriented to guide the design of visual representations that communicate facts that are fixed a priori.

Several others have mostly agreed about the benefit gained from incorporating design patterns into the design process [57][145]. Moreover, in Thomas and Cook's research agenda for visual analytics [132], they call for conducting research to formally define design spaces that capture different classes of visualizations. They further state *"one potential approach is to develop a library of common visualization design patterns from which developers could draw to build new visualizations".* For instance, Stolte et al. [127] introduces design patterns to describe different forms of zooming within multi-scale visualizations. Chen [25] suggests high-level visualization patterns to address general visualization concerns. Sedig and Parsons [115] propose a catalogue of design patterns that support the design of visualization-based computational tools for complex cognitive activities. All of them conclude that, properly used, design patterns help designers make their design simpler, more flexible, modular, reusable, and understandable. Nevertheless, most of them are too tailored to specific visualizations, and therefore are not as effective when applied to other type of visualizations.

Chapter 3. Problem Identification

If you don't make mistakes, you are not working on hard enough problems. And that is a big mistake

- *Frank Wilczek,* Nobel Prize for Physics

The previous chapter provided a review of both descriptive models and existing design material for alarm visualization design. Such review identifies factors affecting recurrent design challenges for alarm visualization design related to different knowledge areas, including Alarm Management, Human Factors, and Visualization Design. However, as aforementioned, no single designer can be an expert in every relevant knowledge area, and becoming proficient may require years of experience. To overcome this situation, research literature on design across application domains proposes to reuse prior design knowledge, which can provide useful starting points for recurrent design problems, serve as references for comparing or explaining new ideas, and provide access to relevant design discussions.

This chapter is divided into two main sections. The first section frames the design space for alarm visualization design by characterizing recurrent design challenges for designing alarm visualizations that have been dereived from the literature research. Then, it describes two different case studies that both explore relevant aspects of this design space and serve to illustrate how such design space operationalizes in practice. This characterized design space will guide the construction of the proposed solution. The second section discusses the application of design knowledge reuse in alarm visualization design. As shown in the previous chapter, design rules in the form of design principles, guidelines, and standards for alarm visualization design are used. These design rules can be too abstract, not comprehensive enough, and loosely coupled, being difficult to be interpreted and applied by non-experienced designers. Therefore, a more comprehensive, generative, and cohesive artifact is required.

3.1 Framing the Design Space

3.1.1 Design Challenges in Alarm Visualization Design

Alarm visualizations for operating control systems are a rich source of information for human operators to maintain the awareness of the state of the process being controlled [93]. This relevance of alarm visualizations poses various recurrent design challenges to designers who need to create effective alarm visualizations with all human operator's activities, capabilities, goals, and needs in mind. Displaying large volumes of alarms with different attributes such as typology, priority, or location, and assisting the human operators' processes of sense making of these alarms in order to understand the state of the process being controlled are some of the design challenges for alarm visualization designers. These fundamental and recurrent design challenges that affect the creation of alarm visualizations are motivated below and discussed in further detail. Table 3.1-1 summarizes these design challenges and the factors affecting them. It also relates these factors to the knowledge area(s) involved.

- **Visual scalability.** One of the biggest challenges in alarm visualization design is how to design alarm visualizations that display large volumes of multi-dimensional alarms. This design challenge is related to the previously reviewed concept of *visual scalability*, which has been widely defined as *"the capability of visualizations to effectively display large data sets, in terms of either the number or either the attributes of individual data elements"* [41]. Four relevant factors affect visual scalability are four:
    - **Operation situation.** During critical events, even minor disturbances, the alarm systems might generate a huge, unmanageable amount of alarms. As different studies on alarm systems from the human operator's perspective have pointed out [85,107,125], during routine events, the operator workload is manageable but during critical events the situation is the reversed. The alarm system is producing lots and lots of alarms, and the human operator's is overloaded. Aiming at overcoming this situation, alarm visualizations should be able to scale in order to be able to handle alarm data sets with hundreds to thousands of elements (both for routine events and critical events involving alarms).

- **Human performance.** Human capabilities to detect and identify alarms are limited. The perception of information requires a certain amount of time, and humans can only hold about 7± 2 units of information at the same time as the different guidelines and standards for alarm visualization design remind us [16, 40, 64]. Because of this, it is important that the designer keep in mind that the total number of alarms and their maximum rate of presentation does not overload the human operator. Designers also need to make sure that alarm visualizations address the *Naturalness Principle* proposed by Norman [94] to design effective visualizations. This design principle states that experiential cognition is most effective when a visual representation closely matches the information being presented.
- **Display size.** As Thomas and Cook [132] already pointed out in their research agenda for visual analytics, most existing visualization techniques are designed for one size display, generally a desktop display. They set as a design challenge the need of creating new visualization methods to allow the analysts to make effective decisions in time-critical situations. In alarm visualization design, the increase in the range of display sizes, from mobile devices for personnel-in-field to high-resolution displays like wall-sized displays in control rooms, poses designers the need of design alarm visualizations that make effective use of different display devices understanding the possibilities and limitations of each kind of display.
- **Visual metaphors.** According to the literature in visualization design [20, 141], visual metaphors are the means by which data characteristics are encoded for visual display. This involves not only selection of a metaphor, such as a bar chart, but also both mapping of data attributes onto visual characteristics of the chosen metaphor, such as bar size and colour. Visual metaphors have to be carefully designed to meet information-processing goals of the human operator. With this aim, basic design principles [94, 135] and theories for visualization design such as Gestalt theories [131] or the seven visual variables proposed by Bertin [10] should be also considered.

- **Sense making.** Making sense of alarm information by human operators when this information is massive or uncertain can be a complex process. Another key design challenge in alarm visualization design, which is intrinsically related to the visual scalability design challenge, is therefore how to assist human operators in the process of sense making of such large volumes of alarm information. According to the previous review, this design challenge is related to the *sense making* conceptualization of SA, which has been formally defined as *"the ability or attempt to make sense of an ambiguous situation. It is the process of creating situational awareness and understanding to support decision-making under uncertainty. It is an effort to understand connections among people, places, and events in order to anticipate their trajectories and act effectively"* [70]. Factors affecting sense making are six:

- **Operation goal.** Mattiason's study [85] identified different operation goals according to the status of the process being controlled. In particular, he identified that during routine events the goal is to optimize and operate the process being controlled in a safe manner. When a minor upset occurs, human operators' job is to bring the process back to normal operation. During a critical event or a major upset she/he is expected to bring the process to the nearest safe state and if disaster threatens, shut it down, and try to limit the consequences. According to the design principles proposed by Endsley for SA [44], alarm information should be organized in terms of the operator's major goals, rather than presenting it in a way that is technology-oriented. Design guidelines and standards for alarm visualization design [16, 40, 64] also recommend that alarm information should be presented in a priority layer where any changes are brought immediately to the operator's attention. In this way, alarm visualizations should be able to help the operator to decide which alarms to deal with when several occur at the same time in a disturbance, and to show especially urgent alarms to the operator.
- **Source of information.** Operators have to gather information from a wide variety of sources to build SA. The collected information is used for building a mental model of the situation, a concept referred to by Endsley [44] as SA. Aiming at supporting human operators to build SA, alarm visualizations should be able to distinguish among several information sources such as direct telemetry from large-scale, distributed control

systems or environmental data covering factors such as weather status or lightning strikes.

- **Level of expertise.** According to the Naturalistic Decision Making (NDM) model [80] on how problem solving is achieved in complex and uncertain real world situations, human operators' performance and formulation of goals are also highly dependent on the human operators' level of expertise. Designers need to design alarm visualizations that distinguish human operators' processing information capabilities according to their level of expertise.
- **Environment.** In keeping with the theories of sense-making for complex and dynamic contexts [47,55], designers should be able to design alarm visualizations that help to human operators not only to understand the distribution in space of objects within a contextual environment but also the interactions among people and events in such environment.
- **Update rate.** The status of the process being controlled can change and, in consequence, alarm information should be updated to reflect such change. Standards and guidelines [16, 40, 64] for alarm visualization design, as well as the design principles proposed by Endsley [44] for SA suggest that alarm visualizations should be able to display trends of changing alarm data values in a readable way for human operators. They should make a distinction between changing values that take exact readings and general alarm trends.
- **Time history.** In relation with the update rate of alarm information, standards and guidelines [16, 40, 64] for alarm visualization design, as well as the design principles proposed by Endsley [44] also suggest that an embedded trend in computer-based information displays should cover enough time and be accurate enough to depict the development of situations that vary from preceding operation situations. In keeping with that, embedded trends in alarm visualizations should be displayed with sufficient resolution in time to ensure that rapidly changing variables can be observed and interpreted.

Design Challenge	Factor	Knowledge Area(s) involved	References
Visual scalability	*Operation situation*	Alarm Management, Human Factors	[85,107,125]
	Human performance	Human Factors, Visualization Design	[16,40,64,94]
	Display size	Visualization Design	[132]
	Visual metaphors	Human Factors, Visualization Design	[10, 20, 94, 135, 131,141]
Sense-making	*Operation goal*	Alarm Management, Human Factors	[16, 40, 44, 64, 85]
	Source of information	Human Factors	[44]
	Level of expertise	Human Factors	[80]
	Environment	Human Factors	[47,55]
	Update rate	Alarm Management, Human Factors, Visualization Design	[16, 40, 44, 64]
	Time history	Alarm Management, Human Factors, Visualization Design	[16, 40, 44, 64]

Table 3.1-1 Characterization of design challenges for alarm visualization design

3.1.2 Case Studies

The empirical research carried out for this research work is arranged into two separate case studies that explore the previously characterized design space. For each case study, a general motivation and overview is presented, followed by discussion of the main insights, as well as how they relates to the previous theoretical formulations.

CASE STUDY 1: ALARM VISUALIZATION DESIGN FOR THE ELECTRIC POWER GRID OPERATION

This case study was motivated by the participation of the author of this research work in the Energos project [46], funded by Spanish CENIT Program 2009. The main goal of this project was to develop information and communication technologies that support new operational and organizational challenges caused by the Smart Grid. One of the main activities of the project was concerned with the operation of the Smart Grid within control rooms. The continuous sensing of the grid state, the increasing interconnectivity and complexity of the infrastructure, and the rising amount of operational information to manage require new visual displays that support new operational models, based not only on identifying breakdowns but also on anticipating and diagnosing them. Fig. 3.1-1 shows an example of a visual display from a real control system used for operating the current electric power grid. It clearly shows that an increasing amount of operational information could affect not only its visual scalability but also the process of making sense of such volume of operational information. In particular, achieving an appropriate Situation Awareness (SA) level by control room operator was establised as an essential factor.

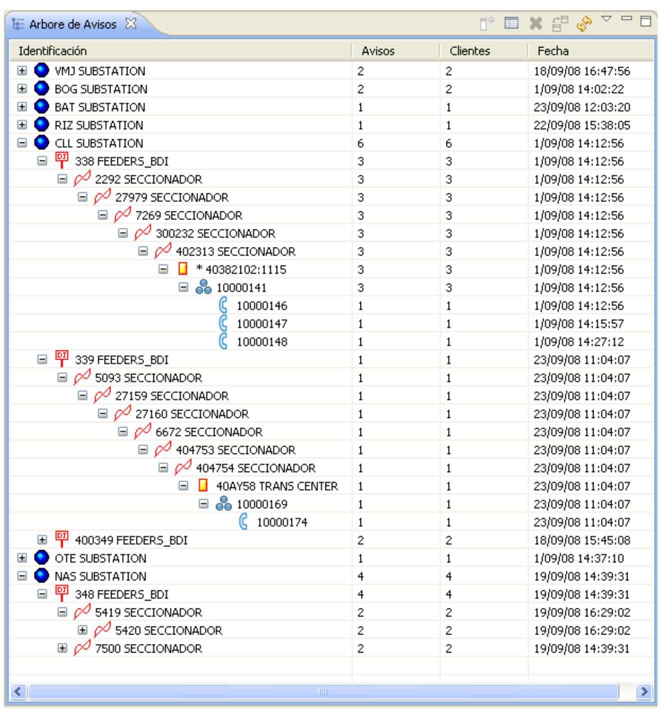


Fig. 3.1-1 Visual display from a real control system used for operating the current electric power grid

It has been widely argued that designing visual displays for SA requires that designers understand how human operators acquire and interpret information as well as identify such factors that underlie this process. Hence, in order to design new visual displays for the Smart Grid operation, a study on the operation of the current grid was firstly carried out, whose results are summarized in [106]. Based on this study, one of the main findings was the characterization of the current grid operation as based on managing alarms. Human operators supervise the performance of planned operations on the grid infrastructure such as discharges in power transformers and detect potential incidents on relevant electrical assets based on the managing of alarms. The control system gives human operators various types of alarms registered by multiple devices and sensors through desktop-sized displays, being the responsibility of the human operator to decide on real time their priority and relevance in order to know the state of the grid. Thus, when a severe disturbance occurs in the grid such as a power outage, there can be many alarms displayed on the desktop displays such that the operators may be overwhelmed and the most important alarms are difficult to locate by them. In contrast to normal operation of the grid, when the goal is to optimize the performance of planned operations, during these situations, the goal is to bring the grid back to a safe manner and to ensure an uninterrupted supply of electricity to critical customers such as data centres or hospitals.

This case study illustrates thus in a real-world context various relevant factors affecting *visual scalability* and *sense making* design challenges for designing alarm visualizations. Regarding visual scalability, two factors are specifically illustrated, including the *operation situation* and the *human performance* in response to alarms. During severe disturbances in the grid, even during planned operations, the alarm systems might generate a huge amount of alarms that human operators may not be able to manage in order to know the state of the grid infrastructure. Regarding sense making, this case study describes two main factors, including the *operation goal* and the *source of information*. Human operators have different operation goals according to the state of the grid infrastructure. In order to know such state, they have to gather alarm information registered from multiple devices and sensors distributed along the grid.

Case Study 2: Alarm Visualization Design for Emergency response

This second case study was motivated by the participation of the author of this research work in the Emercien (Emergency management and civic engagement) project. Emercien is a basic research project funded by the Spanish Ministry of Economy and Competitiveness that aims at deploying sociotechnical platforms that promote effective and reliable collaboration in emergency management among different stakeholders, including first responders, decision and policy makers, volunteers and citizens. One of the main activities of this project was concerned with the participation of emergency communities of volunteers during emergency situations. Emergency communities of volunteers are groups of individuals who altruistically collaborate with official emergency organisms and corps due to their accredited skills and valuable knowledge in specific situations. They make up a monitoring network that tracks emergency alarms related to emergency situations such as a heavy rain emergency alarm declared by emergency managers in an early stage. Volunteers act then as “human sensors”, collecting and sharing information about their evolution. Further details about the characterization of emergency communities of volunteers are described in [58].

In this context, the purpose was to improve the capacity of emergency volunteers to respond to unexpected events through visualization mechanisms that go beyond the current state of research on public participation tools and related technologies. As it was stated in [58], in order to achieve that purpose, these visualization mechanisms should enable collaborative reflection, promote mutual visibility of volunteers’ efforts and sustain a shared view of the community. Similarly, they should facilitate sense making of large, simultaneous and distributed pieces of heterogeneous emergency alarms not only provided by emergency corps or other volunteers but also by common citizens with different levels of credibility and priority. For example, the information coming from social networks is less structured and reliable that the information coming from official emergency organisms and corps. Focused specifically on addressing sense-making issues through visualization mechanisms, different design challenges were identified. The first one was related to the volume of emergency alarms to display. These alarms can vary across emergency situations and be provided from multiple sources such as social networks or mobile devices. Similarly, people with diverse skills and capabilities compose the crowd of citizens who can provide different types of alarms about the emergency situations with different level of credibility. Finally, volunteers also need to track emergency alarms and foresee their evolution across both time and geographical locations in order to support a better response to an emergency situation.

In contrast to the previous one, this case study illustrates diverse relevant factors affecting *visual scalability* and *sense making* design challenges in an emerging context for alarm visualization design. In particular, this case study describes the *operation situation* factor affecting visual scalability. The volume of emergency alarms varies across emergency situations. Regarding sense making, it describes three different factors, including the *source of information,* the *environment*, and the *time history*. Emergency alarms can be provided from different technological platforms such as mobile devices and social networks. Similarly, in order to understand the evolution of an emergency situation, volunteers may require understanding other volunteers' contribution regarding the emergency situation. Finally, this understanding needs to be enriched with assistance for foreseeing the temporal evolution of this emergency situation in order to support a better response.

3.1.3 Synthesis

Throughout sections 3.1.1 Design Challenges in Alarm Visualization Design and 3.1.2 Case Studies, this research work has described the design space for alarm visualization design. It characterizes fundamental and recurrent design challenges that designers need to face when designing alarm visualizations across application domains. From this characterization, it emerges that addressing visual scalability and sense making when designing alarm visualizations requires taking into consideration the role of human operator in response to alarms and the nature of those alarms. It refers to consider human operators' activity as triggered into action by an upcoming alarm. Similarly, alarm visualization designers should recognize the human operator's capabilities, goals and needs. Finally, they should understand the attributes of such alarms and how to support human operators in understanding them through visualization means. Accordingly, alarm visualization design involves having design knowledge from three different knowledge areas, including Alarm Management, Human Factors, and Visualization Design. However, it is difficult for non-experienced designers to understand the constraints and rules that guide the human operator's activities in response to alarms. It is often even more difficult to them to understand the human operator's goals, capabilities, and needs and how to support them through visualization means. One relevant approach to assist designers in such design process is to reuse prior design knowledge for designing alarm visualizations instead of starting from *scratch*.

3.2 Design Knowledge Reuse for Alarm Visualization Design

As shown in the previous chapter, in the context of alarm visualization design, reusing prior design knowledge mostly includes approaches such as design principles, guidelines, standards, and visual languages, which document what are considered the expectations and best practices for designing effective alarm visualizations. The set of existing design design knowledge reuse approaches for alarm visualization design are displayed using a *coloured Venn diagram*, in Fig 3.2-1**.** This diagram also encodes the knowledge areas involved in alarm visualization design as coloured circles, including Alarm Management in turqoise, Visualization Design in light blue, and Human Factors in orange; the variety of descriptive models and perspectives within these areas as lined circles; and existing design rules as rhombuses located according to the perspectives adopted within these areas. In what follows, three fundamental limitations identified for existing design knowledge reuse approaches for alarm visualization design are described:

- **They are not comprehensive enough.** As Fig 3.2-1 shows, there are not design rules that consider all required factors for alarm visualization design, which correspond to the area where the three circles overlap. In particular, the design principles for SA proposed by Endsley can be characterized as the most comprehensive design rules, considering a more extensive range of key design factors than other existing design knowledge reuse approaches for alarm visualization design. However, these design principles do not provide any discussion about how to support human operators in the process of sense making of alarm information, which must depend on designer's experience and circumstances.
- **They are too abstract.** For an abstraction of a reusable design component to be effective, it must express all of the information that is needed by the designer who uses it [76]. This may include space and time characteristics, precision statistics, or scalability limits. In the case of the existing design knowledge approaches for alarm visualization design, they provide a too high-level description of the important characteristics of alarm visualizations and when to use them, but they do not express the how, which must rely again on designer's experience.
- **They are loosely coupled.** To integrate a reusable component into a system effectively, the designer must clearly understand the component's interface; it

means those properties of the component that interact with other components [76]. However, these rules are independent resources that do not provide any description of how they can be combined together in order to generate an alarm visualization. It is a designer's task to understand how to combine them, which may lead not only to an expensive and time-consuming design process but also to create alarm visualizations without sound reasoning and therefore ineffective for supporting human operator's tasks.

In light of these limitations, a new design knowledge reuse approach for alarm visualization design that assists non-experienced designers is required. To this purpose, design patterns have been characterized as the most successful approach to encapsulate reusable design knowledge. Compared to other design knowledge reuse approaches such as design principles or guidelines, design patterns describe proven solution approaches or ways to a solution that, by being applied to different applications and specific contexts of use, lead to new and different solutions. However, they neither offer standardised solutions that can always be used in the same form again nor do they offer new solutions. In other words, a design pattern reflects an abstract solution approach that includes the unchanging components of all solutions that successfully deal with a specific design challenge. The integrated examples of how the solution approach can be applied add clarity and are a help to non-experienced designers in interpreting the guidance. Accordingly, next chapter presents a new approach for reusing previous alarm visualization design knowledge based on the use of design patterns.

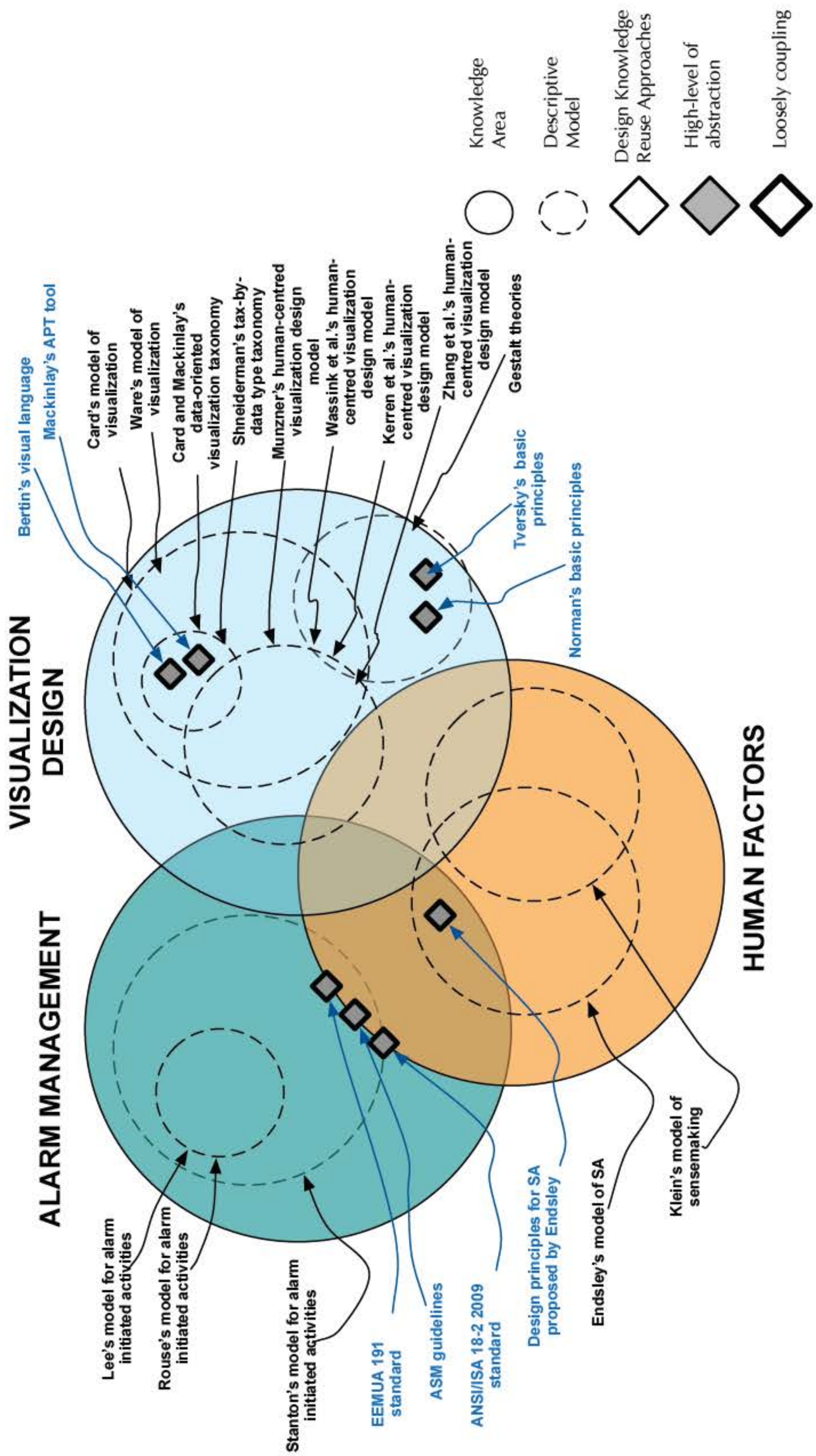


Fig 3.2-1 Venn diagram of design rules adopted for alarm visualization design

Chapter 4. A Design Pattern Language for Alarm Visualization Design

There is nothing new under the sun, but there are lots of old things we don't know yet

- ***Ambrose Bierce,* American Writer**

The previous chapter characterized two fundamental and recurrent design challenges in alarm visualization design, *visual scalability* and *sense making*, and described the limitations of existing design knowledge reuse approaches to be applied by non-experienced designers. These design challenges involve designers having a multi-dimensional knowledge in different knowledge areas including Alarm Management, Human Factors, and Visualization Design. Nevertheless, as previously reported, designers cannot be experts in all areas and becoming proficient may require years of experience [51]. Aiming at overcoming this situation, this research work proposes the use of design patterns as suitable artifacts to facilitate the process of reusing previous design knowledge for alarm visualization design. Compared to other efforts to reuse previous design knowledge, design patterns capture design practice, facilitate multiple levels of abstraction, include the statement of a problem that recurs repeatedly, and deliberately scope their context of application. However, taken in isolation, design patterns are, as Dearden and Finlay [32] firstly stated, *"at best, unrelated good ideas"*. They need to be organized in a meaningful way in order to provide coherent support for design generation.

There are two different levels of organisation for design patterns. On a first level, the design patterns can be organized into collections according to pre-defined criteria. On a second level, a collection of design patterns can evolve and form a language, where each pattern contains backward references to patterns that set its context and forward references to patterns that can be used to help realise the current pattern. A key concept in distinguishing pattern collections from pattern languages is the idea of generativity. It means a design pattern language allows designers to generate designs by implicit sequencing of decisions, derived by traversing the network of links between the individual patterns. It provides a more cohesive structure, with higher-level patterns yielding contexts, which are resolved by more detailed patterns. In keeping with these features, this research work specifically proposes a design pattern language as a fitting approach for facilitating the process of reusing previous alarm visualization design knowledge.

A pattern language can be referred to as *"a collection of patterns which at every level of scale, work together to resolve a complex problem into an orderly solution according to a predefined goal"* [9]. The intention of a pattern language, as described by Alexander [6], is to represent design problems, which makes the problems easier to solve by reducing the gap between the designer's knowledge and the design task. Moreover, Alexander proposes that anyone, not only designers, can use a pattern language as a common language, or a lingua franca. He suggests a pattern language may serve also as a type of boundary object, which may enable communication among people from different disciplines.

This chapter describes the proposed design pattern language in more detail. Firstly, it explains the rationale of the selection of a design pattern language as a fitting design knowledge reuse approach for alarm visualization design. Secondly, the elements of the design pattern language and the design pattern language itself are further described. Finally, its process of usage is depicted.

4.1 The Rationale of the Solution

The design of alarm visualizations defines the way in which the alarm information is presented to human operators, as well as the means by which operators provide inputs to an alarm visualization, receive information from it, and manage the tasks with access and control of alarm information. During this design process, there is a large body of design knowledge that designers call upon and use to match the ever-increasing complexity of design problems. Design knowledge can be generated by observing and experiencing, by interpreting information and data or through reasoning and combining pieces of knowledge [3]. However, not all of these types of design knowledge can be reused. As it was described before, while explicit knowledge can be articulated and more easily transmitted across designers and organizations, tacit knowledge resides within particular individuals in ways that make their actions and decisions difficult to replicate. A third type of knowledge to take into consideration is implicit knowledge, which although is not easily articulated by designers, it can be elicited and articulated by others. Once articulated, explicit and implicit knowledge can be represented as information and thus reused in a consistent and repeatable manner [3]. Other researchers have extended this first distinction of design knowledge according to the focus of concern: process and product knowledge. In the first one, design knowledge is concerned with the activity of designing itself and is both closely tied to the designer who developed it and shared mainly through person-to-person contacts. On the contrary, the product knowledge is concerned with the artifact to be designed. This includes requirements, various kinds of relationships between parts and assemblies, geometry, functions, behaviour, various constraints associated with products, and design rationale. In this way, product knowledge can be characterized as more reusable than process knowledge. Aiming at both representing and articulating design knowledge that can be reused for designing alarm visualizations, this research work is particularly focused on both *explicit* and *implicit product design knowledge.* It includes existing alarm visualization designs in terms of visual features, visual structures and view transformations. To this end, it uses a *design pattern language* approach, which has been considered to be especially well suited for the reuse of design knowledge in a variety of design situations.

A design pattern language provides a framework upon which any design can be anchored [111]. It means that the language does not determine the design. It allows both capturing the essential bits of a problem-solution couple in a specific context, and representing it in a way, *design patterns*, so that it can be applied and adapted in different situations by non-expert practitioners, and even users [6]. Design patterns are not created or invented; they are built via an invariant principle of good design as manifest across different contexts [111]. In particular, they represent the *how* and *why* to solve the design rationale. Design patterns make easier to reuse successful designs and more accessible to designers of new systems. They may already be very valuable for representing design knowledge but when patterns are related to each other, it is when they can reach potentially a far more valuable thing. Such a set of connected patterns is called a design pattern language. By imposing constraints, a design pattern language articulates a large number of possibilities while still allowing an infinite number of possible designs [111]. In this case, the remaining choices are precisely those that connect human beings either visually, emotionally, functionally, or by facilitating their interactions and activities.

4.2 The Design Pattern Language

Defining a design pattern language places a number of challenges. It is firstly required to establish the elements of a design pattern language. Previous work related to the definition of design pattern languages for other purposes such as the design of visualization-based computational tools for complex cognitive activities [115], object-oriented software [51], user interfaces [136], interaction design [137], security systems [56], and hypermedia systems [88] has been considered. However, due to both the completeness and detailed structure, the approach to define the elements of this design pattern language is an integration of the two latter ones. As a result, this design pattern language is composed by the following elements: (i) a *catalogue of design patterns* that captures design practice and embodies knowledge about successful solutions for alarm visualization design; (ii) a *classification scheme* that organizes the collection of design patterns according to different criteria; and (iii) a *design pattern language* as a result of organizing the interrelationships between these designs patterns.

In second place, it is required to identify the adequate source material for the definition of a pattern language. Trying to address the previous characterized design problems in alarm visualization design, this design pattern language results from both an extensive review of design principles, standards and guidelines, controls systems reports, and visualization and interaction techniques from Alarm Management, Human Factors, and Visualization Design, and an evaluation with designers. In particular, card-sorting exercises [18] with designers were conducted. Direct feedback from them was also elicited using a mix-questionnaire. The evaluation outcomes helped to both identify misunderstandings, or ambiguous terminology in order to refine the preliminary design pattern language (see further details of the evaluation in Section 5.3.1 Expert-based Evaluation). Therefore, this language does not have the bias of one person or a group. It spans both the alarm visualization design practice and literature.

Finally, it is needed to determine an appropriate mechanism to document pattern languages. According to Hafiz et al. [56], the relationships between patterns are described through diagrams. In particular, two types of diagrams are mainly used to document pattern languages, including *Alexandrian-style diagrams* and *text-annotated diagrams.* In the Alexandrian style, the patterns are organized in a *directed acyclic graph* (DAG)*,* from most general to most concrete; an arrow from one pattern to another represents a structural or temporal refinement. This style therefore relies on hierarchy and makes it easy for the reader to identify which patterns to consider at any point in the pattern language. The text-annotated diagrams include *textual annotations* on the connections between patterns. They pack more information and give the reader a much quicker overview, but they lack standard meanings or reusable vocabulary for annotations and they trade off the ability to follow patters in a sequence [56]. Considering the lack of reusability of the text-annotated diagrams, this research work applies the Alexandrian style to document this design pattern language for alarm visualization design. The following sections describe all these aspects in more detail.

4.2.1 Building the Catalogue of Design Patterns

Cataloguing is a key step of growing a pattern language in a specific domain [56]. It consists of the selection and classification of a set of suitable design patterns for the purpose at hand. Nevertheless, in keeping with the research literature on pattern languages, there exists different interpretations of what is and isn't a pattern and its abstraction level. There is no objective metric indicating the abstraction level of the problem addressed by patterns [77]. One person's pattern can be another person's primitive building block. According to Seeman [116], design patterns that are abstract are usually ideal for reuse purposes. The more specialized a design pattern gets the more difficult to reuse it. In this work, adapting the widely used definition proposed by Gamma et al. [51] in software engineering, design patterns are referred to as:

Definition 2. Design Patterns. *"Descriptions of features, visual structures, and view transformations to define computer-based interactive visual representations of alarm information".*

This catalogue of design patterns results from an extensive literature review of relevant sources from different areas such as Alarm Management, Human Factors, and Visualization Design. In particular, the literature reviewed can be classified into two broad categories: (i) *grey literature* - informally published written material such as reports on control systems, standards and design principles that may be difficult to trace via conventional channels such as published journals and monographs because it is not published commercially or is not widely accessible; and (ii) *peer-reviewed literature* articles and books on visualization and interaction techniques that have been evaluated by several researchers or subject specialist in the academic community prior to accepting it for publication.

Grey literature	Standards [64][40] and guidelines [16][37] for alarm management	Control systems reports [1][2][5][22][23][28][29][30][38][39][42][43][53][62][65][89][95][98]][99][102][113] [121][123][134][143] [146]			
Peer-reviewed literature	Design principles for human factors [45][67]	Models of visualization design [20][141]	Models of alarm-initiated activities [79][108][125]	Visualization design principles [94][135][10]	Visualization and interaction techniques [20][41][124]

Table 4.2-1 The source material for the catalogue of design patterns

As a result of this review, this catalogue contains ***29 design patterns*** that address the design space for alarm visualization design. The main criterion of including a design pattern in the catalogue is *its application in more than one different situation.* For ease of reading, further characterizations of the design patterns are annexed in Annex A - The Catalogue of Design Patterns for Alarm Visualization Design.

As an example of the selection of a design pattern to be included in this catalogue, *Details hierarchy* pattern results from a variety of sources such as standards on alarm management and control system reports. These sources recognize the design problem across application domains of addressing the exploration of the overall alarm information according to different alarm dimensions in order to support the diagnosis of the cause of a failure in a controlled process. To this problem, all of them propose the use of a standard display hierarchy (see Fig. 4.2-1) to present the multi-level views necessary for exploring alarm information. These levels follow an expected progression from the general to the more detailed in order to aid the operator in performing different tasks. The rationale of this solution is grounded in that a hierarchy delivers a robust structure that encourages ready access to information while at the same time keeping important situation context and promoting efficient navigation to go deeper.

Fig. 4.2-1 Different implementations of the Details hierarchy pattern. On the left side, this design pattern is applied to a control system interface for power plants. On the right side, this pattern is applied to the SIMATIC PCS 7 control system developed by Siemens [121]

4.2.2 Describing Design Patterns

Each design pattern is divided into eight descriptive blocks that provide textual information about several aspects of a design pattern. Specifically, the description of every design pattern follows an adaptation of the format proposed by Alexander et al. [6]. This template lends a uniform structure to the information, making design patterns easier to learn, compare, and use [51]. The meaning of each block is described as follows:

- ***Pattern identifier and name*.** An alphanumeric identifier is assigned to each design pattern. An order number in addition to both the first letter of its purpose category and the first letter of its scope category composes it. For example, a design pattern classified under the Presentation category with Feature level of abstraction is identified as *PF(X). Design Pattern name*. Afterwards, a pattern's name is also assigned to each pattern.
- ***Classification.*** The pattern's classification reflects the scheme introduced in Section 4.2.3 Defining a Classification Scheme.
- ***Context.*** This block describes the design context in which this pattern should be considered. Different solutions can arise from the same design problem occurring in different contexts. Accordingly, this block outlines the set of situations in which the pattern is effective in responding to what the human

operator needs. It also describes any other design patterns that lead to this design pattern.

- ***Problem.*** This block provides a brief description of the design problem that trigger the usage of the pattern in question. It is an expression of what the human operator needs to do, perceive, or understand.
- ***Solution.*** This descriptive block outlines the crucial characteristics of the solution that address the problem identified. It captures details of how the solution can be executed, as well as further clarifying scope and context.
- ***Known uses.*** This block features two different graphic examples that demonstrate the essential characteristics of the design pattern visually.
- ***Rationale.*** This block delivers an argumentation for the usage of a pattern. This argumentation is mostly based on theoretical grounds related to Alarm Management, Human Factors, and Visualization that justify the use of the pattern.
- ***Relations.*** While most patterns can be used as individual entities to perform a certain task autonomously, the basic idea of a pattern language is to connect several modules with each other based on a concrete application context. Following the connective rules for pattern languages proposed by Salingaros [110], the collection of patterns presented in this work uses five types of relations: (1) a pattern is called the *specialization* of another pattern when it shares the same functionality but possesses more specialized characteristics or features; (2) the logical consequence of the specialization pattern is its inversion: the *generalization*. As the name implies, this pattern has rather generic features to serve a more universal purpose; (3) for some design purposes, there is more than one possible solution available. The *alternative* relation makes allowance for these use cases signalling the designer that he can choose from several options that lead, more or less, to the same result; (4) the *combination* relation is an alliance of patterns to be applied together in order to address a specific human operator task; and (5) the *composition* relation indicates that a pattern is composed of one or more other patterns.

4.2.3 Defining a Classification Scheme

Design patterns are numerous and have common properties. Because there are many design patterns for alarm visualization design, it is needed a way to organize them. A classification scheme makes it easy to refer to families of related patterns, to learn the patterns in the catalogue, and to find new patterns [117]. However, there is no standard way of organizing or grouping design patterns. It depends on the needs of designers for addressing specific design problems. Following the same approach than the visual language for web design patterns described at [35] the classification scheme presented here (see Fig. 4.2-2) is organized in *purpose* and *level of abstraction*.

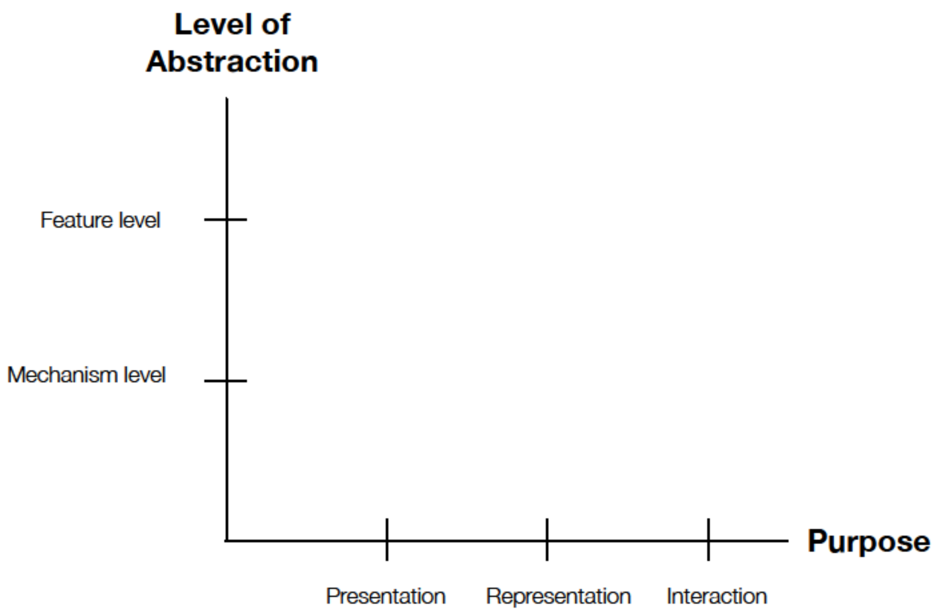


Fig. 4.2-2 Classification scheme

The first criterion, called *purpose*, reflects what a design pattern does. Design patterns can have *presentation*, *representation*, or *interaction* purposes. These purposes have been identified as relevant to most applications of visualization [124]. In particular, *presentation* patterns are related to design decisions about how to organize the alarm information on the interface of an alarm visualization. *Representation* patterns refer to the assignment of specific marks and graphic properties to alarm information attributes. *Interaction* patterns are associated with those graphical parameters that allow the human operator producing a change in a view of a corpus of alarm information, easing the acquisition of insight. Therefore, there will inevitably be interdependence between presentation, representation and interaction patterns. Even if represented alarm information is chosen to be displayed, there remains the question as to whether its display should be under interactive control [124].

The second criterion, called *level of abstraction*, specifies whether the design pattern reflects visual features that an alarm visualization should support, thus, the design pattern is very general, or depicts how these features should be supported. Alarm visualization design literature includes different levels of abstraction. *Feature level* patterns are the more general patterns. They describe visual features that the alarm visualization should support. A feature in this work is a grouping of visual capabilities that provides value to the user. *Mechanism level* patterns are of a detailed nature depicting how the visual features of an alarm visualization should be addressed. In particular, they describe different visual mechanisms such as visual structures and view transformations that should be applied to support such visual features. Visual structures are defined as marks and graphics properties to encode information [20]. View transformations are defined as graphical parameters that interactively modify and augment visual structures [20].

Note, as it shown in Fig. 4.2-2, that the two criteria of this classification scheme can be assumed as orthogonal. Orthogonality means that any of the design patterns classified by purpose may be classified as well by level of abstraction. This classification scheme forms the basis for growing the pattern language. Table 4.2-2 lists all 29 design patterns using this classification scheme.

Level of abstraction		Purpose: Presentation	Purpose: Representation	Purpose: Interaction
	Feature	PF(1) Details hierarchy PF(2) Spatial dedication and continuous visibility PF(3) Primary level of detail PF(4) Secondary level of detail	RF(1) Highlighting RF(2) Visual coding schema RF(3) Integrated displays RF(4) Trend displays RF(5) Lists	IF(1) Direct manipulation IF(2) Display resolution management
	Mechanism	PM(5) Overview and detail PM(6) Tiled layout	RM(6) Brightness RM(7) Flashing RM(8) Colour RM(9) Maps RM(10) Diagrams RM(11) Bar charts RM(12) Histograms RM(13) Pie charts RM(14) Linear charts	IM(3) Dynamic queries IM(4) Brushing and linking IM(5) Zooming IM(6) Panning IM(7) Scrolling IM(8) Paging IM(9) Distortion

Table 4.2-2 Classification of design patterns using the classification scheme

The design patterns classified by *Presentation* purpose with *Feature* level of abstraction are related to both structuring the alarm information into several detailed levels (*PF(1) Details hierarchy, PF(3) Primary level of detail,* and *PF(4) Secondary level of detail);* and presenting the most relevant alarms visible all the time (*PF(2) Spatial dedication and continuous visibility*)*.* Consequently, the design patterns classified by *Presentation* purpose with *Mechanism* level of abstraction specialize these design patterns and specify the arrangement of both these detailed levels *(PM(5) Overview and detail)* and the most relevant alarms (*PM(6) Tiled layout*) on the alarm visualization interface*.*

The *Representation* design patterns with *Feature* level of abstraction are about providing distinctive encodings of relevant alarm information *(RF(1) Highlighting);* providing different visual properties to support the distinction of categories of alarms *(RF(2) Visual coding schema);* integrating alarm information into process displays to represent causal relationships between alarms (*RF(3) Integrated displays*); displaying trends to support the projection of future states of the controlled process *(RF(4) Trend displays)*; and displaying the chronological order of alarms (*RF(5) Lists)*. In this way, *Representation* patterns with *Mechanism* level of abstraction specialize these patterns and propose specific visual mechanisms for both relevant information *(RM(6) Brightness* and *RM(7) Flashing)* and categories of alarms (*RM(8) Colour).* They also describe specific integrated displays for representing both geographical *(RM(9) Maps)* and functional relationships between alarms *(RM(10) Diagrams).* Similarly, they describe specific trend displays, including *RM(11) Bar charts, RM(12) Histograms, RM(13) Pie charts,* and *RM(14) Linear charts.*

Finally, the design patterns classified under the *Interaction* purpose with *Feature* level of abstraction are about allowing the operator to manipulate alarm information on the interface, using actions that correspond at least loosely to manipulation of physical objects (*IF(1) Direct manipulation*); and manage the display space *(IF(2) Display resolution management)*. Consequently, the design patterns classified by *Interaction* purpose with *Mechanism* level of abstraction specialize these design patterns and specify both the actions *(IM(2) Dynamic queries* and *IM(3) Brushing and linking)* and view transformations to deal with a limited display space *(IM(5) Zooming, IM(6) Panning, IM(7) Scrolling, IM(8) Paging,* and *IM(9) Distortion).*

4.2.4 Defining a Design Pattern Language

The previous collection of design patterns captures the essence of the design problems and solutions when designing alarm visualizations. It provides design patterns classified by purpose in combination with level of abstraction, easing the selection of an appropriate solution for the problem at hand. However, as aforementioned, design patterns by themselves are not enough for communicating a holistic view of the design of alarm visualizations. Alarm visualization design defines how the alarm information should be presented to the human operator, the visual displays that make up an alarm visualization interface, the roles played by these displays, their building blocks, and how they interact among them and with the operator. Consequently, design patterns should be combined in a cohesive way in order to support this more general design purpose. To achieve this, this research work provides a design pattern language that aims to make explicit the underlying design knowledge applied to combine these patterns.

The construction of this design pattern language follows the strategy applied by Hafiz et al. [56] to grow a pattern language for security. This strategy is based on the study of the connections among design patterns in the small scale. In particular, following the connective rules for pattern languages proposed by Salingaros [110], this design pattern language displays five types of connections, which are mentioned in the "Relations" section of each pattern between design patterns:

- *Composition relationship* This relationship indicates that a pattern is composed of one or more other patterns.
- *Combination relationship* This *relationship* indicates an alliance of patterns to be applied together in order to address a specific human operator task.
- *Alternative relationship* This *relationship* indicates several design patterns that lead more or less, to the same result.
- Specialization *relationship* This *relationship* indicates when a pattern shares the same functionality than other but possesses more specialized characteristics. The logical consequence of the specialization pattern is its inversion: the *generalization*. This relationship indicates that a pattern has rather generic features to serve a more universal purpose.

According to these connections, three small pattern languages have been created, one for each cluster of patterns corresponding to the *purpose* categories. In each grouping, the patterns were ordered in the typical order they would be applied in practice. Then, these small diagrams were combined into one large diagram, adding some inter-group relationships. The details of this process, and the resulting design pattern language diagram are described in what follows.

Six patterns are listed in the *Presentation* category in Table 4.2-2. Four out of six are patterns classified under *Feature* level of abstraction. Accordingly, the relations between them were firstly reviewed. Naturally, it was started with the structure of the alarm visualization interface *(PF(1) Details hierarchy).* In practice, a designer defines the structure of the interface and compartmentalizes it. This structure can be described as a composition of different levels of detail of alarm information *(PF(3) Primary level of detail* and *PF(4) Secondary level of detail). PF(3) Primary level of detail* describes the need of presenting a first level of detail or overview of the alarm information. *PF(4) Secondary level of detail* refers to provide a secondary level of detail that presents all the alarm information regarding key elements of the controlled process. Therefore, different composition arrows related *PF(1) Details hierarchy* with both *PF(3) Primary level of detail* and *PF(4) Secondary level of detail*. To these different levels of detail of alarm information, a designer should add the presentation of the most relevant alarms that will remain manageable under all controlled process conditions (*PF(2) Spatial dedication and continuous visibility). PF(2) Spatial dedication and continuous visibility* describes the need of presenting the most important alarms, always visible, in a spatially dedicated position. As a consequence, a combination arrow related these two patterns.

After relating these *Feature* level patterns, connections with and between *Mechanisms* level patterns were secondly identified. A way of presenting the different levels of detail of alarm information on the interface is by using *PM(5) Overview and detail* pattern. *PM(5) Overview and detail* refers to simultaneous display of both an overview and detailed views of the alarm information, each in a distinct presentation space. Accordingly, a specialization arrow from *PF(1) Details hierarchy* to *PM(5) Overview and detail* was added. Similarly, *PM(6) Tiled layout* specializes *PF(2) Spatial dedication and continuous visibility. PM (6) Tiled layout* recommends using a tiled structure to present a summary of the most relevant alarms. Thus, a specialization arrow from *PF(2) Spatial* *dedication and continuous visibility* to *PM (6) Tiled layout* was also added. Fig. 4.2-3 shows the resulting pattern language.

Fig. 4.2-3 Pattern language of Presentation design patterns

Fourteen patterns are listed in the *Representation* category in in Table 4.2-2. Five out of fourteen are patterns classified under *Feature* level of abstraction. Accordingly, the relations between them were explored in first place. In practice, after defining the structure of the alarm visualization interface, a designer defines the visual displays formats that make up an alarm visualization interface, the roles played by these displays, and their building blocks. The main visual displays formats can be *RF(3) Integrated displays*, *RF(4) Trend displays,* and *RF(5) Lists. RF(3) Integrated displays* show alarms in a consistent way across controlled process information. They should ensure that most important alarms are easily distinguishable *(RF(1) Highlighting and RF(2) Visual distinction)*. Therefore, different combination arrows related these five patterns. Similarly, they should also provide both interaction mechanisms *(IM(3) Dynamic queries)* and display resolution mechanisms to ensure human operator's tasks *(IM(5) Zooming, IM(6) Panning,* and *IM(9) Distortion*). It was also found that *PF(3) Primary level of detail* can incorporate a combination of separate alarm visual displays formats (*RF(5) Lists*) and integrated displays (*RF(3) Integrated displays*). However, these inter-group relationships between design patterns were not considered until the construction of the final design pattern language diagram. *RF(4) Trend displays* show trend data to ensure that rapidly changing variables in the controlled process can be interpreted. They can be part of both primary *(PF(3) Primary level of detail)* and secondary levels of detail of alarm information (*PF(4) Secondary level of detail*). These inter-group relationships were added to the final design pattern language diagram. Finally, *RF(5) Lists* shows alarm information arranged in a chronological order. As *RF(3) Integrated displays*, they should provide both interaction *(IM(4) Brushing and linking)* and display resolution mechanisms *(IM(7) Scrolling and IM(8) Paging)* to ensure human operator's tasks. Nevertheless, as above, these inter-group relationships were only considered at the final design pattern language diagram.

Based on the relationships identified between *Feature* level patterns, connections with and between *Mechanism* level patterns were reviewed in second place. Two types of integrated displays can be distinguished, including *RM(9) Maps* and *RM(10) Diagrams. RM(9) Maps* pattern displays geographical relationships between alarms. *RM(10) Diagrams* pattern displays functional relationships between alarms. Therefore, a specialization arrow from *RF(3) Integrated displays* to both *RM(9) Maps* and *RM(10) Diagrams* related these design patterns. Likewise, there exists different types of trend displays such as *RM(11) Bar charts*, *RM(12) Histograms*, *RM(13) Pie charts*, and *RM(14) Linear charts. RM(11) Bar charts* pattern presents a display in which numeric quantities are represented by the linear extent of parallel lines, either horizontally or vertically. *RM(12) Histograms* pattern describes a type of bar chart used to depict the frequency distribution for a continuous variable. Therefore, a specialization arrow from *RM(11) Bar charts* to *RM(12) Histograms* was added. *RM(13) Pie charts* pattern presents a circular chart that represents magnitude or frequencies. *RM(14) Linear charts* pattern represents relationships between two or more continuous variable. As a result, a specialization arrow from *RF(4) Trend displays* to *RM(11) Bar charts*, *RM(12) Histograms*, *RM(13) Pie charts*, and *RF(14) Line charts* was added. Similarly, two alternative techniques to highlight critical information can be used, including *RM(6) Brightness* and *RF(7) Flashing. RM(6) Brightness* highlights relevance by making an object appear brighter than others. *RM(7) Flashing* increases salience by increasing and decreasing in alteration the brightness of an object or its background. An alternative arrow was added thus to represent the alternative relation between these two patterns. To support the distinction of alarms (*RF(2) Visual distinction*), colour coding can be used to code categories of alarms (*RM(8) Colour*). Therefore, a specialization arrow from *RF(2) Visual distinction* to *RM (8) Colour* was added. Moreover, *RM(6) Brightness* is a colour feature. Consequently, a specialization arrow from *RM(8) Colour* to *RM(6) Brightness* related these two patterns. Fig. 4.2-4 shows the resulting pattern language.

Fig. 4.2-4 Pattern language of Representation design patterns

Nine patterns are listed in the *Interaction* category in Table 4.2-2. The relationships between them were explored together. Firstly, designing the interaction features for an alarm visualization involves defining the type of dialogue through which the operator and the system interact and the type of actions that the operator can perform on a display. Particularly, it was identified that this dialogue should allow operators to act on visible objects to accomplish tasks (*IF(1) Direct manipulation*). Two different implementations of this dialogue are *IM(3) Dynamic queries* and *IM(4) Brushing and linking. IM(3) Dynamic queries* allow operators explore different subsets of the alarm information by manipulating selectors. *IM(3) Dynamic queries* is mostly combined with *RF(3) Integrated displays. IM(4) Brushing and linking* creates a tightly coordination between alarm visual displays under the selection of specific items in a display or set of displays. Therefore, *IM(4) Brushing and linking* can be combined with different alarm visual displays formats *(PM(5) Overview and detail).* However, due to the inter-group nature of these relationships between patterns, only a specialization arrow from *IF(1) Direct manipulation* to *IM(3) Dynamic queries* and *IM(4) Brushing and linking* was added. Secondly, designing the interaction features for an alarm visualization also involves considering the display space issues when alarm information is too large to be displayed all at once with a level of resolution adequate for operators' tasks *(IF(2) Display resolution management)*. *IF(2) Display resolution management* describes the need of using mechanisms that allow the operator specifying a different degree of interest in different parts of information. There exist different specific mechanisms to deal with this lack of display space, including *IM(5) Zooming*; *IM(6) Panning*; *IM(7) Scrolling*; *IM(8) Paging;* and *IM(9) Distortion*. *IM(5) Zooming* and *IM(9) Distortion* can be characterized as alternative mechanisms to allow an operator to obtain details of a selected portion of a large visual display. IM(6) Panning allows moving a viewing frame over a display space of greater size. It was found that these mechanisms are particularly combined with visual display formats that integrate alarm information into process displays (*RF(3) Integrated displays*). However, this inter-group relationship between design patterns was not considered until the construction of the final design pattern language diagram. *IM(7) Scrolling*, and *IM(8) Paging* patterns are alternative solutions to allow the operator to move across alarm information that does not fit the display. These techniques were also found particularly suited to different visual display formats, including *RF(5) Lists*. However, as in the previous case, these inter-group relationships between design patterns were not considered until the construction of the final design pattern language diagram. In this way, only alternative arrows between these design patterns were included. Fig. 4.2-5 shows the resulting design pattern language.

Fig. 4.2-5 Pattern language of Interaction design patterns

Previous pattern languages shown in Fig. 4.2-3, Fig. 4.2-4, and Fig. 4.2-5 are relatively self-contained, because they describe solutions for different classes of alarm visualization design problems. However, different inter-group relationships were identified during the construction of these pattern languages. Hence, in order to create the final diagram, shown in Fig. 4.2-6, these additional relationships have been added. Here are the main additional connections made for this diagram:

- Providing a first level of detail of alarm information *(PF(3) Primary level of detail)* combines *RF(3) Integrated displays, RF(4) Trend displays* and *RF(5) Lists*.
- Providing a secondary level of detail of alarm information *(PF(4) Secondary level of detail)* requires the use of *RF(4) Trend displays*.
- Interacting with alarm information that does not fit the display space involves combining *IM(5) Zooming, IM(6) Panning,* and *ID(9) Distortion* with *RF(3) Integrated displays*. Similarly, *RF(5) Lists* should be combined with *ID(7)Scrolling*, and *ID(8) Paging*.

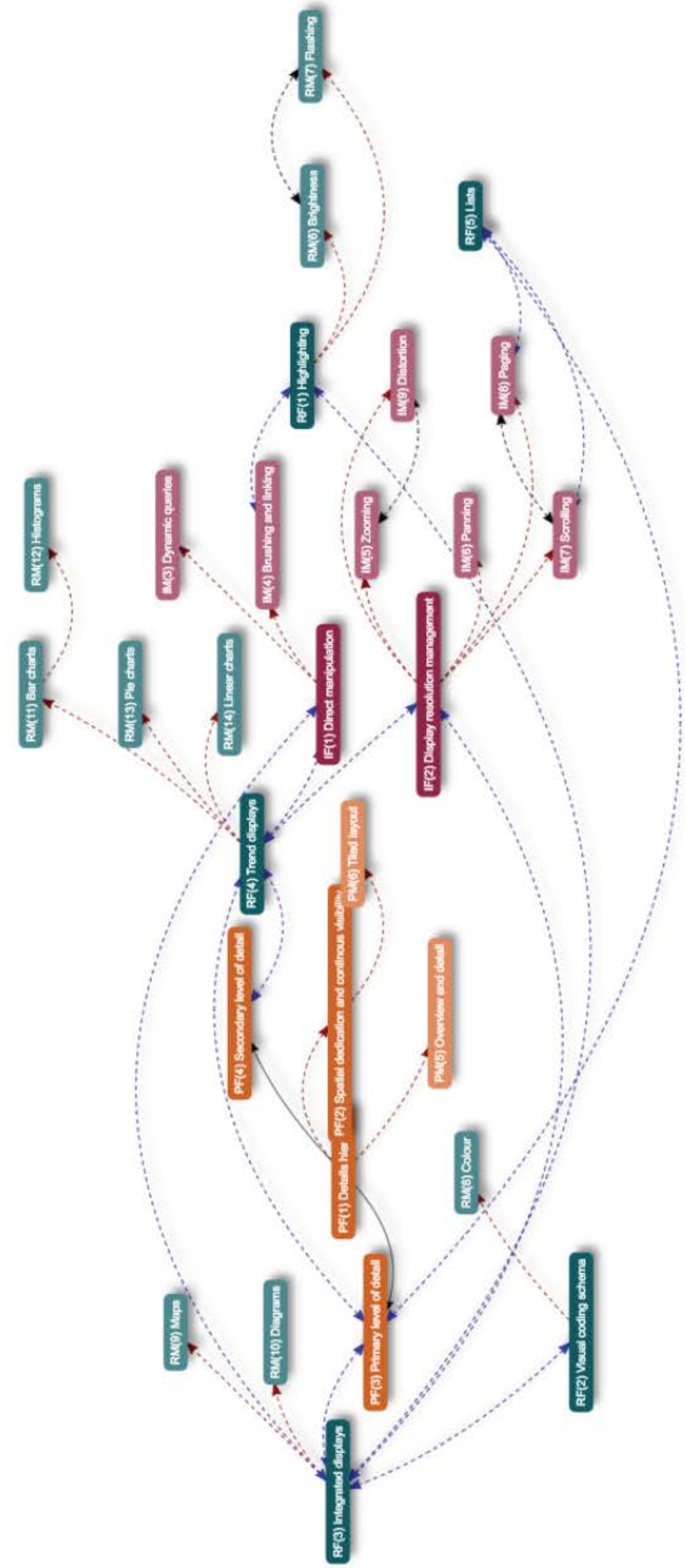


Fig. 4.2-6 Pattern language final diagram

4.3 Using the Design Pattern Language

This design pattern language represents the essential design knowledge for designing alarm visualizations. It seeks to facilitate the design tasks by complementing existing design methods and processes with descriptions of recognized alarm visualization design solutions that have developed and evolved over time. As it has been already defined, the visualization design process can be thought of as adjustable mappings from data to visual form to the human perceiver. However, the most important mappings in this process are both the transformation from *data tables* to *visual abstractions* and the transformation from *visual abstractions* to *views* [20]. Visual abstractions are defined as *"structures that combine values and available vocabulary of visual elements"* [20]. Visual abstractions can be further transformed by *view transformations* by adding controls for interaction.

Under the human-centred visualization design approach, these mappings are not only driven by data characteristics but also by the domain problem characterization. In particular, it is required the characterization of the user's tasks at different levels of abstraction for making visual encoding and interaction decisions (see Fig. 4.3-1).

Fig. 4.3-1 Integration of the human-centred visualization design approach into the visualization reference model

It is at this step in which design patterns become relevant design tools to inform the transition between these structures (see Fig. 4.3-2). They take into account human operator tasks and supply visualization and interaction design solutions that support these tasks.

Fig. 4.3-2 Integration of the design patterns into the visualization design process

Following a similar approach than in the process flow diagram proposed by [88], Fig. 4.3-3 shows a process flow diagram that depicts the integration of the design patterns into the alarm visualization design process. Following, each of the steps of this process is further described.

Step 1. At this first step, a designer must elicit a set of design requirements in terms of both human operator tasks and information requirements in a target environment. Interviews and other ethnographic methods can be used to achieve this purpose.

Step 2. This second step consists of mapping the design requirements elicited from the vocabulary of the environment into a more abstract and generic description that is the vocabulary provided by the design patterns. To achieve that, the catalogue of design patterns defined in Section 4.2.1 Building the Catalogue of Design Patterns) supports the designer. The classification of design patterns by *purpose* and *scope* supports this mapping.

Step 2.1. *Eliciting requirements is not easy, even when a designer has access to target users [265]. There can be situations in which the previous mapping cannot be performed due to an incomplete or too general requirement characterization. Similarly, the designer can also make a mistake in this mapping process. In both cases, it is necessary to return to the Step 1.*

Step 3. The third step is the process of refining the previous requirements mapping, taking into account the existing relationships between design patterns defined in the design pattern language diagram (see Fig. 4.2-6). This refinement can be seen as an iterative process to which the design pattern language provides a framework upon which the design solution can be anchored.

Step 3.1. *Based on the content of the "Relations" field of a design pattern, it is required to follow the "composition relationships" of a design pattern in question. It means that others patterns compose a design pattern, which should be included for completeness.*

Step 3.2. *It is also required to consider all those design patterns that establish a "combination relationship" with the pattern at hand. They can be applied together in order to address a specific requirement.*

Step 3.3. *Similarly, it is needed to consider all those design patterns that establish an "alternative relationship" with the pattern at hand as a potential design alternative.*

Step 3.4. *Finally, regarding "specialization relationships", it is needed to consider if the design pattern in question should be replaced. In that case, it is necessary to return to the Step 3.*

Step 4. The fourth step aims at organizing the previous mappings according to the corresponding purpose, including *presentation, representation, and interaction*. Each design pattern in turn may have different level of abstraction, categorized into *Feature*, and *Mechanism*.

Step 5. Finally, the solution should be created. Previous design patterns should be instantiated considering the particularities of the application domain. As a result, an alarm visualization design should be generated.

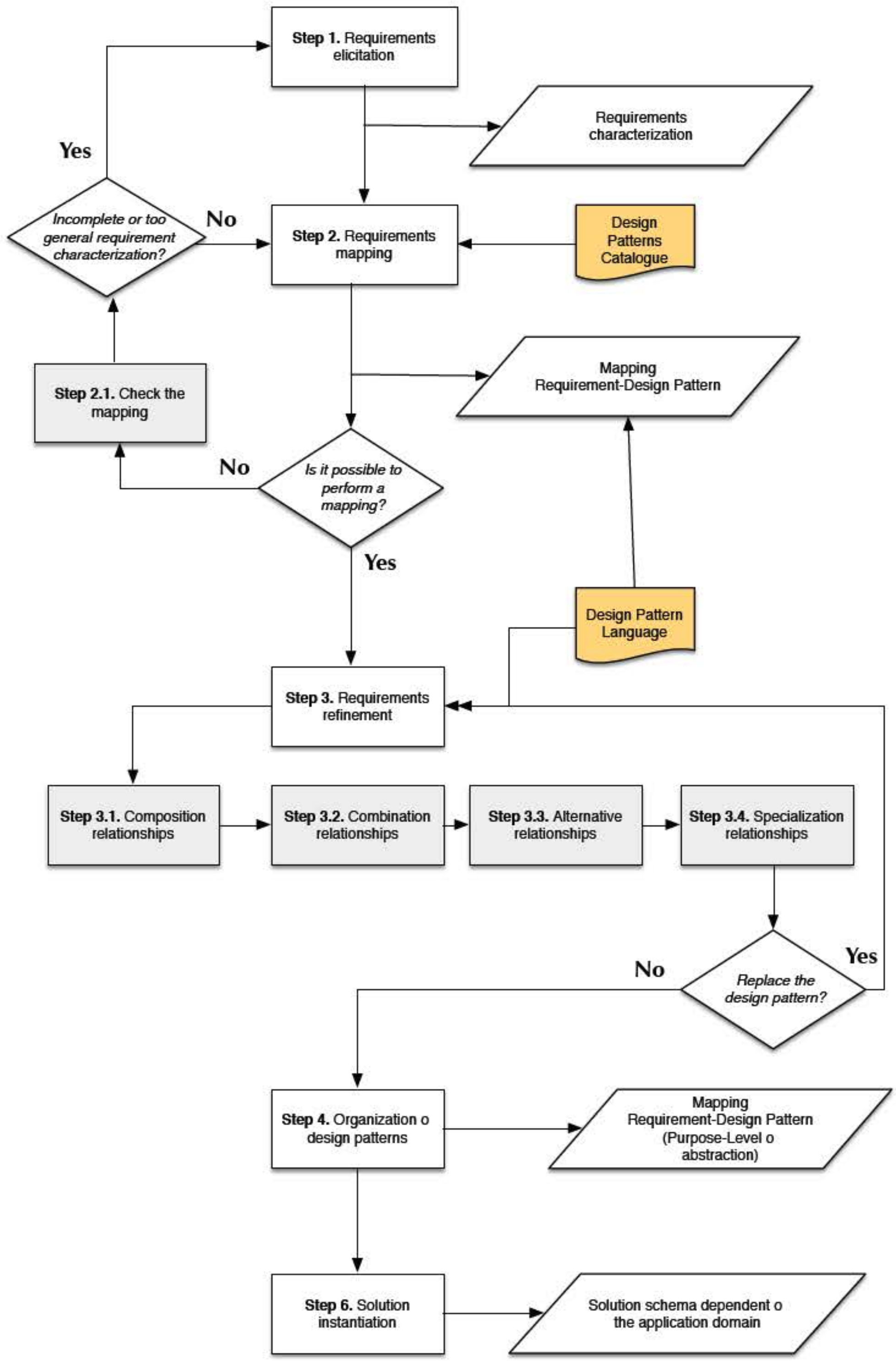


Fig. 4.3-3 Process flow diagram of the integration process of the design patterns into the alarm visualization design process

Chapter 5. Evaluation

Everything that can be counted does not necessarily count; everything that counts cannot necessarily be counted

- *Albert Einstein,* German Physicist

Evaluation is a crucial component of the research process [59]. It is focused on assessing that a proposed solution meets the requirements identified in a research work. Evaluation requires the definition of goals, the definition of appropriate metrics based on such goals, and the gathering and analysis of appropriate data to validate their achievement. As a consequence, each evaluation is quite specific to the artifact being evaluated, its purpose, and the goals of evaluation. In design science research in the area of Information Systems (IS), evaluation is particularly concerned with examining design science research outputs, including design artifacts and design theories [138]. It provides evidence that a new designed artifact achieves the purpose for which it was designed. Without evaluation, outcomes are unsubstantiated assertions that the designed artifacts, if implemented and deployed in practice, will achieve their purpose. Rigorous, scientific research requires evidence.

This chapter presents the evaluation process of the design pattern language for alarm visualization design and its results. In particular, it firstly describes the evaluation plan for successfully conducting the evaluation of the proposed solution in reference to the goals of this research work. In second place, it explains the evaluation conducted, emphasizing both the type of evaluation methods utilized and the results obtained. Finally, it discusses the analyses of such results.

5.1 Evaluation in Research

Evaluation is defined as *"the systematic determination of merit, worth, and significance of something or someone"* [59]. Evaluation theory has its roots in social inquiry and the desire for accountability and control. Depending upon the objectives of the evaluation, different methods and strategies are used to guide inquiry. The methods and strategies are selected, in turn, based on the type of evaluation required. Particularly, evaluation falls into three main types, those oriented toward the construction of knowledge, those oriented toward placing value, and those oriented toward how an artifact is used [7]. Evaluation can be further broken into two distinct categories, *formative* and *summative* [114]. Formative evaluation focuses on processes and summative evaluation focuses on outcomes. Based on such distinctions, next section depicts a set of common steps that characterizes every evaluation process. Afterwards, it depicts the goals of this evaluation. Finally, the description of the evaluation methods used is provided.

5.1.1 The Evaluation Process

Irrespective of when, where, and how an evaluation is done, all evaluation studies have certain structure in common. That structure is described in what follows:

- *To position the evaluation.* Positioning the evaluation involves determining the boundaries of the evaluation, and defining the set of objectives that will be assessed. Objectives guide the evaluation by helping to determine its scope.
- *To plan the evaluation.* Having identified the objectives of the evaluation, the next step is to choose what data is needed to answer the evaluation questions, how the data will be analysed, which evaluation methods will be used and how the results will be presented. This latter choice depends on what data is needed to answer the questions and which theories are appropriate to the context.
- *To collect data.* The data collection task mostly consists of gathering data, which has three main parts, including determining the source of information, developing data collection instruments such as interview guides and questionnaires and collecting the information.
- *To analyse and synthesize data*. Analysing and synthesizing data provide ways of discerning, examining, comparing and contrasting, and interpreting meaningful patterns or themes. Meaningfulness is determined by the particular objectives of the research work at hand; the same data can be analysed and synthesized from multiple angles depending on the particular research or evaluation questions being addressed.

- *To communicate evaluation findings.* In the final stage of an evaluation, it is required to communicate to the appropriate parties the findings that are based on the previous syntheses.

5.1.2 The Goals of the Evaluation

Following Hevner et al. [59] many researchers in the field of design science research have argued for *quality* and *utility* of design artifacts. Researchers state that design artifacts can be evaluated in terms of functionality, completeness, consistency, accuracy, performance, reliability, usability, fit with the organization, and other relevant quality attributes, whereas utility is often the defining characteristics of artifact evaluation. As a consequence, and according to Gamble and Goble [50], it is possible to differentiate between quality and utility in the form of dependent and independent evaluation respectively: (i) quality – *a function of the artifact or process assessed against a quality specification to provide a specific, objective measure of quality*; and (ii) utility – *a function of the artifact and user to assess whether the output fits the purpose and meet the needs.* The interpretation of these evaluation parameters within the context of this research work is further described in what follows and summarized in Table 5.1-1 and Table 5.1-2.

Quality. In keeping with the research literature on pattern languages, when discussing the quality of a pattern language, different parameters related to both the language and the design patterns need to be considered. In particular, *internal connectivity* [110] is identified as a desirable attribute of a pattern language. *Generativity*, *descriptive*, *recurrence*, and *explicative* refer solely to design patterns [86]. The *internal connectivity* of a pattern language is concerned with its *integrity* and *maturity*. The integrity of a design pattern language can be defined as the organization of patterns into a hierarchy where the higher-level patterns provide a conceptual description but also provide the context in which the lower level patterns could be used. Links within levels indicate that a language is developing maturity as readers can better understand a language if it has organization at different levels. Similarly, several attributes can affect the quality of the resulting collection of patterns. *Generativity* refers to how well the patterns teach to build their manifestations. *Descriptive* is concerned with the capability of patterns of describing the nature of the solution proposed. *Recurrence* refers to the reappearance of the design pattern in different situations. Finally, *explicative* is related to how well the design pattern arguments the reasons for its application.

Evaluation Goal	Evaluation Parameters				
		Parameter	Definition	Sub-parameters	Definition
Quality	*Design Pattern Language*	Internal connectivity	Connectivity between the levels in the language's hierarchy [110]	*Integrity*	Organization of patterns into a hierarchy where the higher-level patterns provide a conceptual description but also provide the context in which the lower level patterns could be used [133]
				Maturity	Existence of several entry points to the most appropriate level of patterns [133]
	Design Patterns	Generativity	A good pattern teaches how to build their manifesta ions. It should be possible to visually depict he kind of structure that results from a pattern application, at least in general or schematic terms [86]	No applicable	No applicable
		Descriptive	The pattern must describe the nature of the solution, at least in a general way [86]	No applicable	No applicable
		Recurrence	A pattern must be a recurring phenomenon [86]	No applicable	No applicable
		Explicative	A good pattern gives a sense of the reasons why the solution is appropriate [86]	No applicable	No applicable

Table 5.1-1 Summary table of the interpretation of the quality of the design pattern language

Utility. The essential aim in design science research is to rigorously demonstrate the utility of a design artifact being evaluated. Rigor should be approach by establishing if the artifact solves the stated problem and causes and observed improvement, its efficacy [59]. This latter can be specifically referred to as *"the degree to which the designed artifact produces its desired effect considered narrowly, without addressing situational concerns"* [138]. In accordance to that, the utility of the design pattern language for alarm visualization design is related to its *operational feasibility*, *usability*, and *efficacy*. The operational feasibility of this design pattern language can be understood as the capability of the design pattern language of being used to design alarm visualizations for operating control systems. The usability of an artifact is described in terms of *ease of use*, *ease of learnability* and *level of satisfaction* [66]. Finally, the efficacy of a design pattern language refers to that attribute that is reflected by, and embodied in, their sense of audience [133]. For this research work, there is one audience, non-experienced designers. This design pattern language will be effective if non-experienced designers are able to reuse design knowledge items for alarm visualization design.

<table>
<tr><th>Evaluation Goal</th><th colspan="6">Evaluation Parameters</th></tr>
<tr><td rowspan="6">Utility</td><td rowspan="6">Design Pattern Language</td><td>Parameter</td><td>Definition</td><td>Sub-parameters</td><td>Definition</td></tr>
<tr><td rowspan="3">Usability</td><td rowspan="3">The ease of use and learnability of the design pattern language</td><td>Satisfaction</td><td>The comfort and acceptability of use</td></tr>
<tr><td>Ease of use</td><td>The extent to which the design pattern language can be used by non-experienced designers to accomplish an alarm visualization design</td></tr>
<tr><td>Ease of learnability</td><td>The ease of accomplishing alarm visualization design tasks the first time they encounter the design pattern language</td></tr>
<tr><td>Operational feasibility</td><td>The capability of the design pattern language of being used to design alarm visualizations for operating control systems</td><td>No applicable</td><td>No applicable</td></tr>
<tr><td>Efficacy</td><td>The degree to which the design pattern language allows non-experienced designers reusing alarm visualization design knowledge</td><td>No applicable</td><td>No applicable</td></tr>
</table>

Table 5.1-2 Summary table of the interpretation of the utility of the design pattern language

5.1.3 The Evaluation Methods

The selection of evaluation methods must be matched appropriately with the kind of designed artifact being evaluated and the selected evaluation goals [59]. In accordance to that, previously to select the evaluation methods to be used in this research work, it is required to review the nature of the evaluand and the evaluation goals identified. Regarding the nature of the evaluand, based on the literature, Venable et al. [138] distinguishes two different classification of designed artifacts: (1) *product artifacts* from *process artifacts*; and (2) *technical artifacts* and *socio-technical artifacts.* Product artifacts are technologies such as tools, diagrams, or software that people use to accomplish some task. Process artifacts are methods, and procedures that guide someone or tell them what to do to accomplish some task. Technical artifacts are those that do not require human use once instantiated. Socio-technical artifacts are ones with which humans must interact to provide their utility. This research work proposes a design pattern language as a process artifact that guide non-experienced designers in the design of alarm visualizations. It can be also characterized as a socio-technical artifact since designers must interact with it in order to solve their design problems.

Regarding the evaluation goals, as it was depicted in Section 5.2.2 The Goals of the Evaluation, the evaluation of this design pattern language is focused on demonstrating its *quality* and *utility* to allow non-experienced designers to reuse previous alarm visualization design knowledge. In particular, to demonstrate the quality of the design pattern language, a set of different dimensions related to both the pattern language and the design patterns need to be evaluated. Similarly, to demonstrate the utility of the design pattern language, its operational feasibility, usability, and efficacy need to be assessed. By considering these aspects, it is possible to identify a set of possible evaluation methods that fit the nature of the design pattern language and the evaluation goals. In what follows, the selected evaluation methods are described.

- **Expert-based evaluation.** An expert-based evaluation utilizes the knowledge of professionals in a specific area to evaluate a designed artifact. The purpose of the expert-based evaluation of the design pattern language is twofold. On the one hand, it uses the knowledge of designers to identify misunderstandings, or ambiguous terminology in order to refine the definition of the design pattern language. On the other hand, it allows validating the design patterns according to the agreement of designers over its quality. It is based on the use of both a mix-questionnaire and card sorting exercises.

- **Analytical evaluation.** The purpose of an analytical evaluation is to perform a prospective review of the context of a designed artifact in order to determine its intrinsic qualities. The analytical evaluation of the design pattern language examines its *internal connectivity*. The *internal connectivity* of a pattern language is concerned with its *integrity* and *maturity*. This evaluation is based on the use of the static analysis method.

Table 5.1-3 shows the mapping of the expert-based and analytical evaluations to the quality evaluation goal.

Evaluation Goals	**Evaluation Parameters**	**Evaluation Parameters**	**Evaluation Parameters**	**Evaluation methods**	**Evaluation metrics**
Quality	*Design Pattern Language*	Internal connectivity	*Integrity*	Analytical Evaluation – *Static Analysis*	Representation of a pattern language as a Directed Acyclic Graph (DAG)
			Maturity		Relationships between design patterns establish a hierarchy within the pattern language
	Design Patterns	Generativity		Expert-based Evaluation – *Open-ended questionnaire* *Card sorting method*	Acceptance rate
		Descriptive			
		Recurrence			
		Explicative			

Table 5.1-3 Mapping of quality parameters to evaluation methods

- **Descriptive evaluation.** The purpose of a descriptive evaluation method is to support or refute parts of a designed artifact by means of borrowing the form of design narratives. This descriptive evaluation of the design pattern language constructs an argumentation around the *operational feasibility* of the design pattern language. This evaluation is based on the use of a scenario method.

- **Experimental evaluation.** The purpose of an experimental evaluation is to assess a designed artifact within controlled settings. The experimental evaluation of this design pattern language demonstrates the *usability* and *efficacy* of the design pattern language to non-experienced designers in reusing previous design knowledge for alarm visualization design. It is based on the use of the controlled experiment method.

Table 5.1-4 shows the mapping of the descriptive and experimental evaluations to the utility evaluation goal.

Evaluation Goals	Evaluation Parameters		Evaluation methods	Evaluation metrics
Utility	*Design Pattern Language*	Usability	Experimental Evaluation – *Controlled experiment*	Task level satisfaction
				Ease of use's ratings
				Ease of learnability's ratings
				Usage of design patterns
				Usage of elements of the design pattern language
		Operational feasibility	Descriptive Evaluation– *Scenario method*	Description of the steps carried out to accomplish the design of alarm visualizations for operating control systems
		Efficacy	Experimental Evaluation – *Controlled experiment*	Completeness of generated designs
				Quality of generated designs

Table 5.1-4 Mapping of utility parameters to evaluation methods

5.2 Evaluation Results

The resulting selection of evaluation goals and methods constitute a high-level description of an evaluation process [138]. Based on this high-level description, the specific detailed evaluations must be designed. In this research work the evaluation is composed by the following four activities: (1) an *expert-based evaluation*; (2) an *analytical evaluation*; (3) a *descriptive evaluation*; and (4) an *experimental evaluation with two rounds*. The following sections describe these activities and the results obtained in more detail.

5.2.1 Expert-based Evaluation

An expert-based evaluation utilizes the knowledge of professionals in a specific area to evaluate a designed artifact [59]. In this case, this evaluation uses the knowledge of researchers with experience in design to both identify misunderstandings, or ambiguous terminology in order to refine the design pattern language and validate the quality of the design patterns. To these purposes, *card-sorting exercises* [18] and a *mix-questionnaire* [4] are used. Card sorting is a method used to understand users' expectations and understanding of a specific topic. A mix-questionnaire combines both open-ended questions and closed-ended questions. Open-ended questions enable the respondents to highlight the issues that they find most relevant. Closed-ended questions make it possible to validate some quality attributes of the design patterns by using a four-level summated rating scale ranging from *strongly disagree* to *strongly agree.*

Participants

Seven researchers with experience in design (three women and four men) performed this evaluation. They had experience in *visualization design*, *interaction design*, *software design*, and *web design*. The range of experience went from two to more than six years. All of the participants were familiar with the use of design rules. Table 5.2-1 summarizes the experience of the group of designers who participated in this evaluation.

Experience	Visualization design	Interaction design	Web design	Software design
From 2 to 6 years	100%	71,4%	71,4%	100%
More than 6 years	0%	28,6%	28,6%	0%

Table 5.2-1 Expert-based evaluation: Percentage of designers' years' experience

Methodology

Each participant's session was conducted individually and consisted of two main steps. One researcher directed all these sessions and assisted participants throughout. The first step focused on asking the participant to sort 29 textual cards of pattern descriptions according to pre-defined categories (see Fig. 5.2-1).

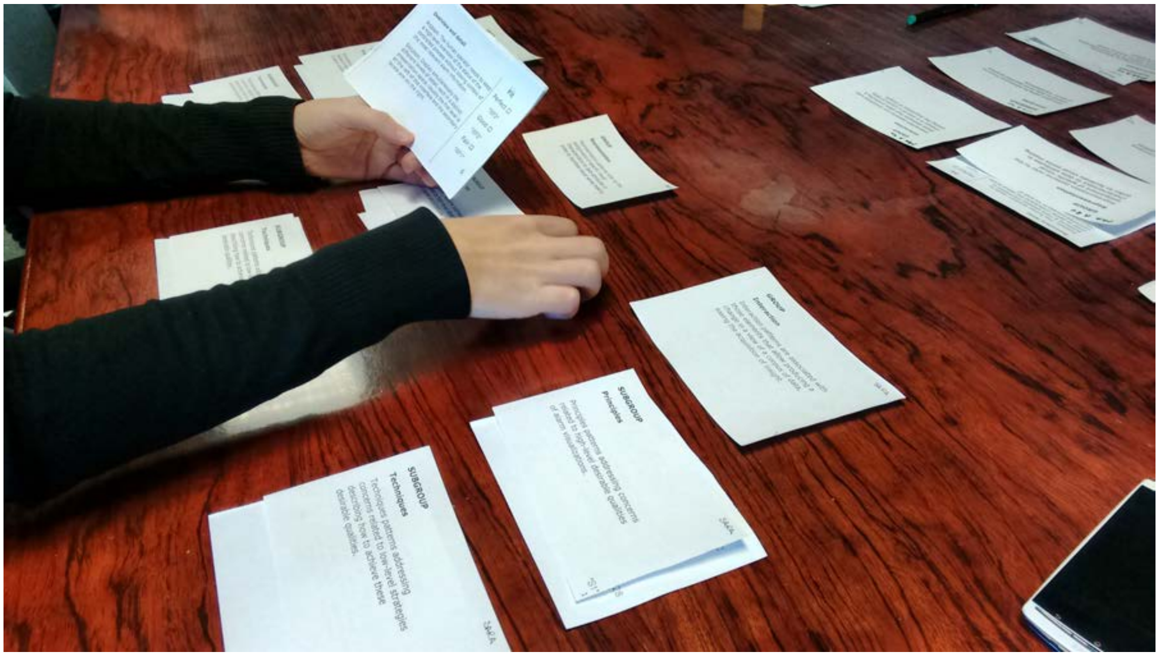


Fig. 5.2-1 Expert-based evaluation: participant sorting textual cards of pattern descriptions.

Each textual card contained a pattern name, pattern problem description and pattern solution description. An overview of the meaning of the pre-defined categories was provided. Once the cards were sorted into groups, the participant was asked to look at each card and mark its quality of fit in the category he/she selected for it: *poor*, *fair* or *perfect*. The participant could also propose new categories labels that make more sense to them if required. Therefore, a combined card sorting method was used. This combination helped to see both how well the pre-defined categories labels worked and how designers grouped the design patterns. This step typically took around 35 minutes.

In the second step, a mix-questionnaire including three open-ended questions and two closed-ended questions was given to the participants. These questions (see Table 5.2-2) were related to both the terminology used to refer and categorize design patterns and the quality of design patterns descriptions throughout the language. A four-value Likert scale was used to collect the opinion of designers in closed-ended questions: *strongly agree* (4), *agree* (3), *disagree* (2), and *strongly disagree* (1). This scale was chosen to avoid neutral responses.

Mix-questionnaire		
Open-ended questions		
Q1	Does the categories labels and terminology used to refer design patterns make sense to you?	
Q2	Do you consider that some of the categories labels do not reflect their purpose?	
Q3	Do you consider pattern names are not expressive enough according to the pattern description?	
Closed-ended questions		
Q4	Do you consider patterns descriptions teach how to build alarm visualization designs?	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q5	Do you consider patterns descriptions are descriptive enough?	☐ 1 ☐ 2 ☐ 3 ☐ 4

Table 5.2-2 Expert-based evaluation: Mix-questionnaire for participants

Analysis and results

The results obtained from the card sorting exercises were analysed using Syncaps v3 [130], a card-sorting analysis package. In card sorting, cluster analysis is used to decide which items are most frequently grouped together by participants. This package allows automatizing it. In particular, an *item dendogram* and a *pairs map* were generated to display the results of this cluster analysis. The item dendogram shows the distances between patterns using a similarity matrix. The more times a pattern is sorted together with another pattern, the more similar they are. They then appear closer in proximity in the dendogram. The clear result from the item dendogram (shown in Fig. 5.2-2) is that a group of seven design patterns classified under different categories within the catalogue (within the cyan band) tended to be sorted together by the seven participants. These design patterns describe design decisions mostly related to both limited display space issues and navigation issues. This grouping could possibility be because most of design patterns related to display space issues involve interaction capabilities. This interactive nature seems to be more relevant for designers to identify these design patterns than the need of handling limited display spaces.

Fig. 5.2-2 Expert-based evaluation: Item dendogram card sorting results

The pairs map (shown in Fig. 5.2-3) also supported this result. This map shows the frequency with which every possible pair of items appeared together in the same groups. The map uses colour saturation to display this so that a white cell shows a pairing that did not occur while a dark cell shows a pairing that was made by most participants. For instance, *Trend displays* pattern and *Diagrams* pattern tended to be sorted together by most participants. As expert data is used in this analysis, the central portion of each cell indicates the expert pairings and alignment. An example of that is the pair of patterns *Primary level* and *Secondary level*. The bottom row shows relative alignment. In this row, darker cells represent items that were consistently grouped while lighter cells indicate less agreement between participants. Considering the results supported by these two diagrams, a refinement of the preliminary version of the catalogue of design patterns was carried out. This refinement consisted of the movement of *Direct manipulation*, *Zooming*, *Scrolling*, *Paging*, *Panning*, and *Distortion* patterns from the *Presentation* category to the *Interaction* category.

Fig. 5.2-3 Expert-based evaluation: Pairs map card-sorting results

Finally, the participants' responses to the mix-questionnaire were analysed. The set of participants' responses to the open-ended questions is gathered in Table 5.2-3. These responses suggested that some dimensions of the classification scheme were ambiguous or misleading. For instance, *Principles* and *Techniques* categories were considered difficult to distinguish. Therefore, some changes and additions in the preliminary version of the classification scheme were conducted, such as the replacement of *Principles* and *Techniques* categories for *Feature-level* and *Mechanism-level* categories. The final version of the elements of the design pattern language is shown on the Chapter 4. A Design Pattern Language for Alarm Visualization Design.

Participants' responses	
***Q1.* Does the categories labels and terminology used to refer design patterns make sense to you?**	
R1.	"It is kind of complex to understand the difference between Principle and Technique categories"
R2.	"It seems the terminology used makes sense to me"
R3.	"I don't understand the difference between Principle and Technique categories"
R4.	"Yes, the terminology and categories allowed me to categorize design patterns"
R5.	"The terminology used seems fair. It was difficult sometimes to mark the quality of fit of the pattern. I would rather something like adequate or not instead of poor, fair and perfect"
R6	"The terminology sounds adequate to me. Sometimes, the names of the design patterns were too broad"
R7.	"The most difficult part to me was understanding the difference between Principle and Technique categories"
***Q2.* Do you consider that some of the categories labels do not reflect their purpose?**	
R1	"As I mentioned before, the difference between the categories Principles and Techniques is not clear to me"
R2	"Yes, the categories Principles and Techniques"
R3	"The purpose of categories is clear but not the purpose of the subcategories that might be pretty subjective"
R4	"The categories are pretty clear. I would explain better the Interaction category describing that is the result of a human operator's action"
R5	"I doubted also about the Presentation and Representation categories. I also doubted about to classify patterns under Principle or Technique category. Most of times

	the design pattern seemed to fit better under the Technique category"
R6	"I think the categories were pretty clear"
R7	"Well, as I mentioned in the previous question, I had some problems distinguishing between Principle and Technique"
***Q3.* Do you consider pattern names are not expressive enough according to the pattern description?**	
R1	"The names are related to the content of the pattern"
R2	"The names makes sense to me. Some of them are too long"
R3	"I would replace the name Tables by Lists"
R4	"The names are very representative of the solution provided by the design pattern. I haven't had problems with that"
R5	"The names in combination with the description of the pattern helped me to understand better"
R6	"The names are adequate to me"
R7	"I would shorten some names of the patterns. They are too long"

Table 5.2-3 Expert-based evaluation: Participants' responses to the open-ended questionnaire

Similarly, the set of participants' responses to the closed-ended questions are also gathered in Table 5.2-4. Since summated rating scales do not provide concrete values but categories, this study used the median to identify the agreement of participants: the design patterns will be considered as valid if the median of participants' opinion is equal or higher than three (agreement level). According to the judgment of participants, the median of agreement score for these questions was set to *three* to Q4 and *four* to Q5. Due to the median of all indicators has achieved the agreement level, design patterns' quality was considered as validated.

	Participants							
Questions	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Median**
Q4	3	4	3	3	4	3	3	3
Q5	4	4	3	3	4	4	4	4

Table 5.2-4 Expert-based evaluation: Results and final score of each participant to the closed-ended questionnaire

5.2.2 Analytical Evaluation

The purpose of an analytical evaluation is to perform a prospective review of the context of a designed artifact in order to determine its intrinsic qualities [59]. The analytical evaluation of this design pattern language uses the *static analysis* method. The static analysis of a designed artifact is based on examining structure of the artifact for static qualities [59]; in this case, these static qualities correspond with the dimensions that characterise design pattern languages. Many researchers have argued that in order to determine the quality of a pattern language, its internal connections should be evaluated. In particular, according to Salingaros [110], to achieve this purpose, it is required to determine how patterns combine to form higher-level patterns containing new information; how linked patterns exist on different levels or how to find patterns in a new language. To address these needs, six questions have been proposed by Todd et al. [133] as forming the basis of tests that can be applied to a pattern language. In particular, the first four questions are particularly intended to validate the *integrity* of a pattern language, while the last two ones seek to determine its *maturity*. These tests have been already used to evaluate a variety of collections of user interface and HCI design patterns. Given the nature of visualizations as effective communication artifacts between the user and the computer, this research work also applies these tests to evaluate the design pattern language proposed. The test questions are:

- **Test 1.** *Do the reference and context links between the patterns form a map?* The context of a pattern can be defined as those patterns that reference it. Consequently, the references are the inverse function of the context. In the design pattern language proposed in Annex A - The Catalogue of Design

Patterns for Alarm Visualization Design context and reference links are identified in the "Relations" block of each design pattern.

- **Test 2.** *Does the context map match the reference map?* Using the results of the previous test, this question refers to the creation of both a context map and a reference map. A comparison of these two maps should show if they match.
- **Test 3.** *Can the map be ordered into a hierarchy of levels?* This question is related to the possibility of organizing design patterns in a hierarchy in which higher-level patterns provide a conceptual description of an alarm visualization system as well as a definition of the context in which lower-level design patterns should be applied.
- **Test 4.** *Can the levels be used to describe an alarm visualization system at different degrees of granularity?* This question is related to the *semantic granularity* provided by the design pattern language*.* Semantic granularity addresses the different levels of specification of an entity in the real world.
- **Test 5.** *How rich are the links within each level of the hierarchy?* This question refers to the definition of the number of links between nodes at each level in the hierarchy of design patterns. To that end, the six-point scale defined in [133] is applied: (*none*) no observed intra level links in any level of the hierarchies identified; (*unknown*) so many links the user becomes confused; (*minimal*) less than 10% of the links are intra level links; (*developed in one level*) at least a third of the reference links within only one level are intra level links; (*developing*) more than 10% of the links are intra level links and occur in more than half of levels below the root; and (*rich*) more than 30% of the links are intra level links and occur in more than half of the levels below the root.
- **Test 6.** *Can the patterns be organised by different classification systems thereby providing alternative viewpoints?* It refers to the existence of different classification criteria for the same collection of design patterns. The classification schema applied for this design pattern language is described in Section 4.2.3 Defining a Classification Scheme

In what follows, the application of these tests to the design pattern language proposed is described.

As it is shown in Fig. 4.2-6, a language map was created from the links mentioned in the "Relations" block of each design pattern. Following the notion of a pattern language proposed by Borchers [12], this map represents this language as a *directed acyclic graph* (DAG), where nodes are patterns and edges describe references and context from a pattern to another. In particular, the set of edges leaving a node is called its references. The set of edges entering it is called its context. The design pattern language passes therefore **Test 1**. To evaluate if the context map matches the reference map, another language map showing context and reference links was constructed. To construct this map, context and reference links were distinguished as follows:

- Context links are shown as dotted lines with arrows.
- Reference links are shown as dashed-lines with arrows.
- Links identified in both patterns are shown as solid-lines with arrows.

In Fig. 5.2-4 due to the context and reference map matched solid lines represent all links. Therefore, the design pattern language passes **Test 2.**

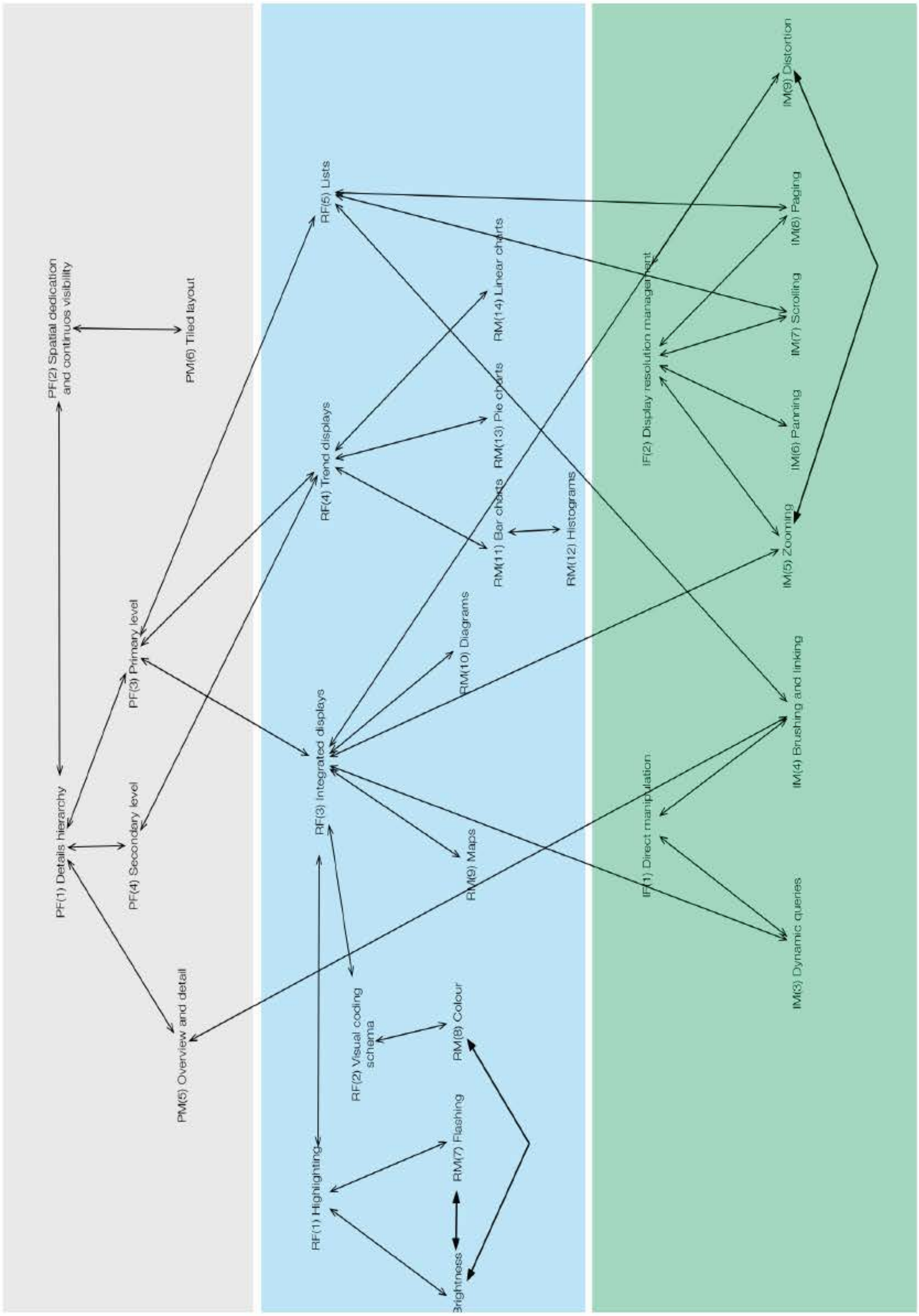


Fig. 5.2-4 Analytical evaluation: Language map showing context and reference links

Examining this language map indicates that there is a pattern, *PF(1) Details hierarchy,* which acts as a root node for a hierarchical structure. In particular, it can be characterized as a three-level hierarchy. The first level of the hierarchy (shaded in grey in Fig. 5.2-4) describes the need of structuring the alarm information into several detailed levels, which can be subdivided into more specific design purposes. The second level of the hierarchy (shaded in blue in Fig. 5.2-4) describes the set of basic representation required to assign graphic properties to alarm information attributes. Finally, the third level of the hierarchy (shade in green in Fig. 5.2-4) describes the set of basic interaction techniques needed to allow the operator to manipulate alarm information on the interface. Consequently, this collection of patterns passes **Test 3.** This design pattern language is also appropriate to pass **Test 4.** This design pattern language provides design patterns with different levels of abstraction as it is described in Section 4.2.3 Defining a Classification Scheme. According to Todd et al. [133], these four passed tests confer to this collection of design patterns the status of **design pattern language.**

The links between patterns within the levels of the hierarchy were also examined. According to Salingaros [110], a pattern language's maturity, is determined by both inter and intra-level links. To that end, a language map was created from the intra-level links mentioned in the "Relations" block of each design pattern. As it shown in shown in Fig. 5.2-5**,** 26 out of 35 of these links are intra-level links. As a result, based on the six-point scale previously described for **Test 5**, this design pattern language can be defined as 'rich'. Finally, although only one hierarchy can be traced through the patterns of the patterns collection, two classification criteria have been proposed (see Section 4.2.3 Defining a Classification Scheme*),* by *purpose* and by *level of abstraction*. Consequently, this language passes **Test 6.** According to Todd et al. [133], these two passed tests confer this language the status of **a mature language.** However, as is expected in such a young discipline as visualization design, this language can't be described as complete. Over time this design pattern language should include new design patterns for alarm visualization design.

Fig. 5.2-5 Analytical evaluation: Language map showing intra-level links

5.2.3 Descriptive Evaluation

The purpose of a descriptive evaluation is to support or refute parts of a designed artifact by means of borrowing the form of design narratives [59]. In this case, the *scenario* method is used to construct a detailed argumentation around the *operation feasibility* of the design pattern language. In particular, two different scenarios have been defined within the context of the two case studies presented in 3.1.2 Case Studies. These scenarios are framed then within two different application domains of alarm visualization design, including *the electric power grid operation* and *the emergency response.* In this way, these scenarios span, respectively, classic and emerging domains where alarm visualizations are used. By applying the design pattern language to these two different contexts; it is possible to ascertain that it can be used to design alarm visualizations for operating control systems across application domains. In what follows, these scenarios are described in further detail. They are described following each of the steps defined in the integration process of the design pattern language (see Section 4.3 Using the Design Pattern Language).

Scenario 1: Designing Alarm Visualizations for the Electric Power Grid Operation

This first scenario is framed within the context of the Smart Grid operation. It describes the application of the design pattern language to an operational user interface based on managing alarms for the Smart Grid. This interface was developed as a part of the Energos project, already described in 3.1.2 Case Studies.

As a result of **Step 1 (Requirements elicitation),** the requirement characterization outlined below is taken and adapted from [106]. This operational user interface seeks to support the most significant operating tasks such as *monitoring* and *controlling* the grid. Monitoring tasks refer to review the status of the grid in order to detect and register potential incidents. Controlling tasks are focused on supervising the performance of planned operations and managing both events and incidents. This interface gives various types of alarms generated by a variety of devices about the status of the grid, being the responsibility of the human operator to decide on real time their priority and relevance. Focusing on the representation of these alarms generated, a set of main design requirements were identified:

- ***RQ1.*** When a severe disturbance occurs, control room operators need to be able to trade with huge pieces of alarm information, which are collected from different devices and with different levels of priority and relevance.
- ***RQ2.*** Control room operators need to make sense of alarm information in relation to the grid connectivity and geographical position.
- ***RQ3.*** They also need to get an overview of incidents and critical events grouped by time, type and devices.
- ***RQ4.*** Control room operators require organizing and searching alarm information depending on the operating situation.

Based on this requirements characterization, **Step 2 (Requirements mapping)** consists of mapping these requirements into a more abstract and generic description that is the vocabulary provided by the design patterns. The use of the catalogue of design patterns defined in Section 4.2.1 Building the Catalogue of Design Patterns facilitates this process. Table 5.2-5 shows the resulting mapping.

Requirement	Design Pattern	
	Classification	Name
RQ1	• Purpose: Presentation • Level of abstraction: Feature	PF(1) Details hierarchy
RQ2	• Purpose: Representation • Level of abstraction: Feature	RF(3) Integrated displays
RQ3	• Purpose: Presentation • Scope: Feature	PF(3) Primary level of detail
RQ4	• Purpose: Interaction • Level of abstraction: Principle	IF(1) Direct manipulation

Table 5.2-5 Descriptive evaluation: Requirements mapping for the first scenario using the catalogue of design patterns

Rooted in this mapping, **Step 3 (Requirements refinement)** was repeated until most of requirements were mapped to design patterns classified under *Mechanism* level of abstraction. Fig. 5.2-6 shows the generated route by performing this refinement process. It is important to point out some of the main decisions made during this requirements refinement process. In order to address **RQ1**, *PF(1) Details hierarchy* was firstly specialized to *PM(6) Overview and detail* pattern and used both *PF(2) Spatially dedicated, continuously visible displays*, which was in turn specialized to *PM(7) Tiled layouts,* and two levels of detail of alarm information (*PF(3) Primary level of detail* and *PF(4) Secondary level of detail)*. In order to address **RQ2,** *RF(3) Integrated displays* was specialized to both *RM(10) Maps* and *RM(11) Diagrams*. *RM(10) Maps* was combined with *RM(5) Colour, IM(5) Zooming,* and *IM(9) Distortion. RM(5) Colour* was, in turn, replaced by *RM(6) Brightness*. To address **RQ3**, *PF(3) Primary level of detail* was specialized to *RF(5) Lists* and *RF(6) Trend displays*, which was specialized to *RM(11) Bar charts*. RM(12) Histograms in turn, replaced this latter pattern. *RF(5) Lists* was also combined with *IM(7) Scrolling* to deal with large volume of alarms. To conclude, aiming at addressing **RQ4**, *IF(1) Direct manipulation* was replaced respectively by *IM(3) Dynamic queries,* and *IM(4) Brushing and linking.* These patterns are used in combination with *RM(10) Maps and RF(5) Lists.*

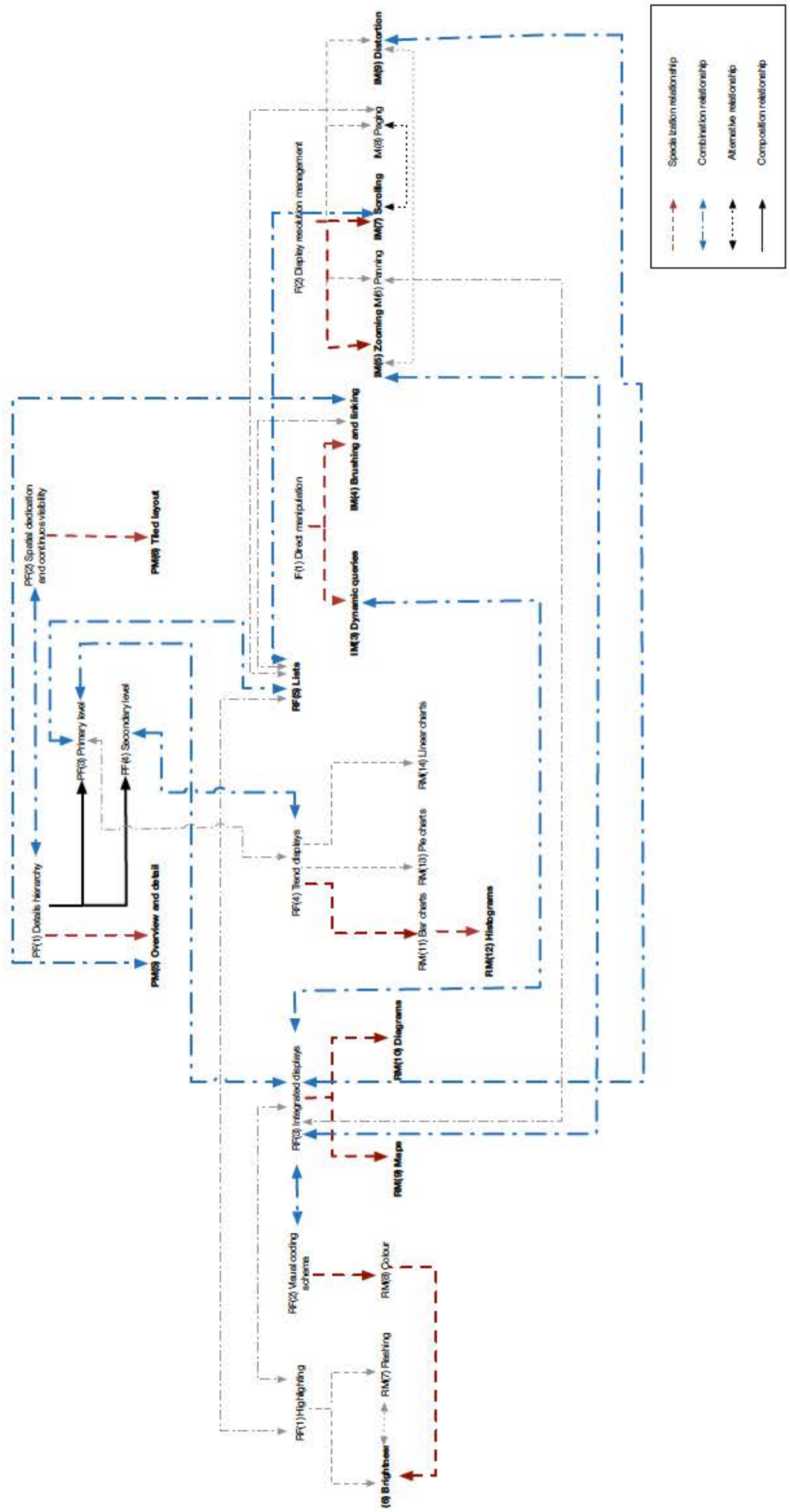


Fig. 5.2-6 Descriptive evaluation: Generated route by the refinement process in the first scenario

Next, **Step 4 (Organization of design patterns)** aims at organize the previous mapping according to the corresponding *purpose*. Accordingly, Table 5.2-6 lists the pair *requirement-design pattern* organized by *purpose*.

Presentation	
RQ1	PM(6) Overview and detail PM(7) Tiled layout
Representation	
RQ2 RQ3	RF(5) Lists RM(6) Brightness RM(9) Maps RM(10) Diagrams RM(12) Histograms
Interaction	
RQ3 RQ4	IM(3) Dynamic queries IM(4) Brushing and linking IM(5) Zooming IM(7) Scrolling IM(9) Distortion

Table 5.2-6 Descriptive evaluation: Organization of design patterns by purpose in the second scenario

Step 5 (Solution instantiation) is related to the definition of the final design solution dependent of the application domain. Accordingly, Fig. 5.2-7 shows how these design patterns were instantiated to develop the final design solution. Further details about this final design solution are described in [106].

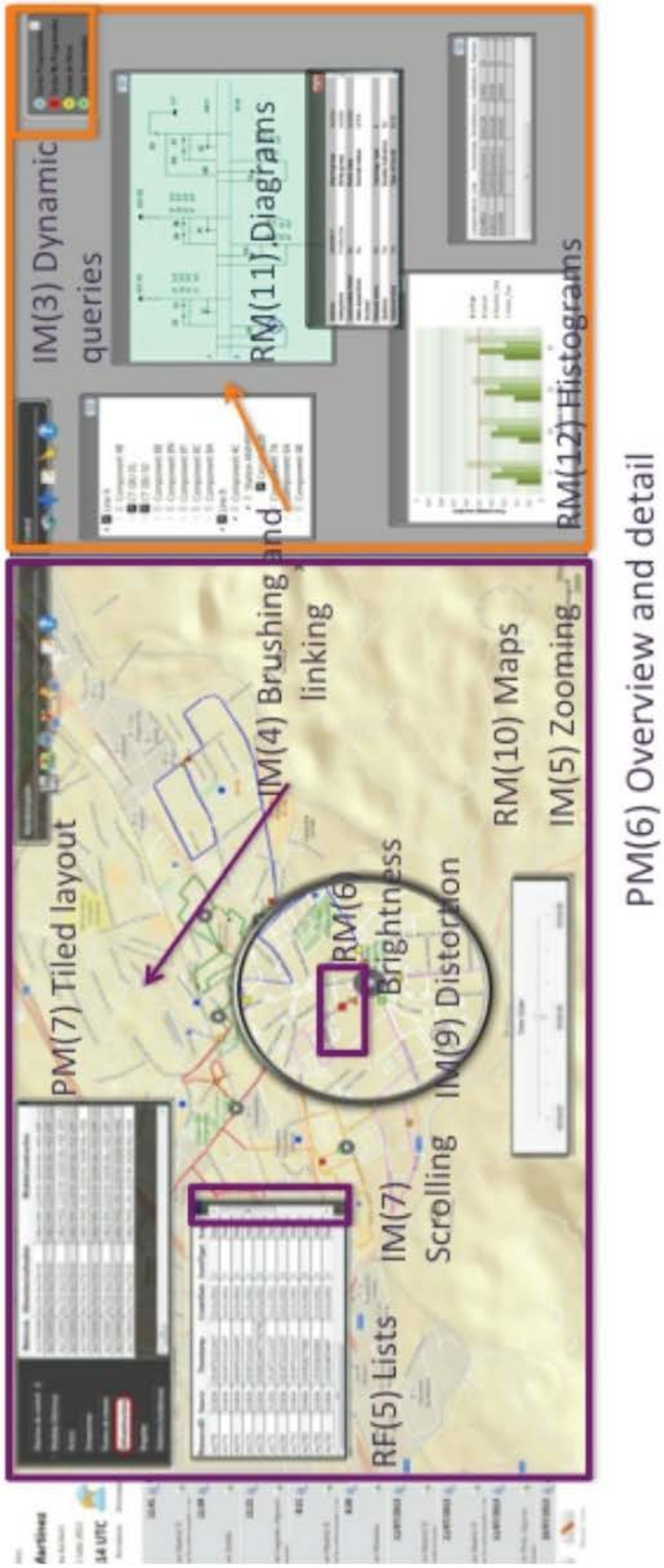


Fig. 5.2-7 Descriptive evaluation: Solution instantiation in the first scenario

SCENARIO 2: DESIGNING ALARM VISUALIZATIONS FOR EMERGENCY RESPONSE

As a result of the **Step 1 (Requirements elicitation),** the requirement characterization outlined below is taken and adapted from the research work described in [58]. This research work was developed within the context of the Emercien project (Emergency management and civic engagement). The Emercien project has been already described in 3.1.2 Case Studies. Focusing on the representation of these alarms, a set of design requirements has been identified:

- ***RQ1.*** The volume of alarms generated can vary across emergency situations. Accordingly, an emergency volunteer needs to be able to explore different volumes of alarms without loosing the track of the overall emergency situation.
- ***RQ2.*** Community volunteers need to track emergency warnings declared by official emergency organisms and corps and foresee their evolution across both time and geographical locations in order to support a better response to a situation.
- ***RQ3.*** People with diverse skills and capabilities compose the crowd of volunteers and citizens who can provide different types of alarms with different levels of priority through different technological platforms. Thus, an emergency volunteer needs to distinguish different levels of alarm priority and diverse alarm sources. Particularly, it is needed to allow the distinction of three different levels of priority, including *high-priority level*, *middle-priority level*, and *low-priority level; and* two main alarm sources, including *volunteer-generated alarms* and *citizen-generated alarms.* This latter source of alarms can be subdivided in turn into alarms reported by citizens from *social networks* and citizen alarms from *mobile applications*.
- ***RQ4.*** Not all emergencies require the same degree of response or attention, and each incident should be evaluated on a case-by-case basis. Therefore, it is required to provide flexible navigation and interaction to emergency volunteers across alarm information.

Based on this requirements characterization, **Step 2 (Requirements mapping)** consists of mapping these requirements into a more abstract and generic description that is the vocabulary provided by the design patterns. The use of the catalogue of design patterns defined in Section 4.2.1 Building the Catalogue of Design Patterns facilitates this process. Table 5.2-7 shows the resulting mapping.

Requirement	Design Pattern	
	Classification	**Name**
RQ1	• Purpose: Presentation • Level of abstraction: Feature	PF(1) Details hierarchy
RQ2	• Purpose: Representation • Level of abstraction: Feature	RF(3) Integrated displays RF(4) Trend displays
RQ3	• Purpose: Representation • Level of abstraction: Feature	RF(2) Visual coding schema
RQ4	• Purpose: Interaction • Level of abstraction: Feature	IF(1) Direct manipulation

Table 5.2-7 Descriptive evaluation: Requirements mapping for the second scenario using the catalogue of design patterns

This previous mapping captures the essence of the design problems and solutions when designing alarm visualizations for emergency communities of volunteers. However, these design patterns by themselves are not enough for communicating a holistic view of the final solution. Rooted in this mapping, **Step 3 (Requirements refinement)** is the process of refining this set of design patterns, taking into account the existing relationships between design patterns defined in the design pattern language diagram (see Fig. 4.2-6). This refinement process can be seen as an iterative process to which the design pattern language provides a framework upon which the design solution can be anchored. This process was repeated until most of requirements were mapped to design patterns classified under *Mechanism* level of abstraction. Fig. 5.2-8 shows the generated route by performing this refinement process.

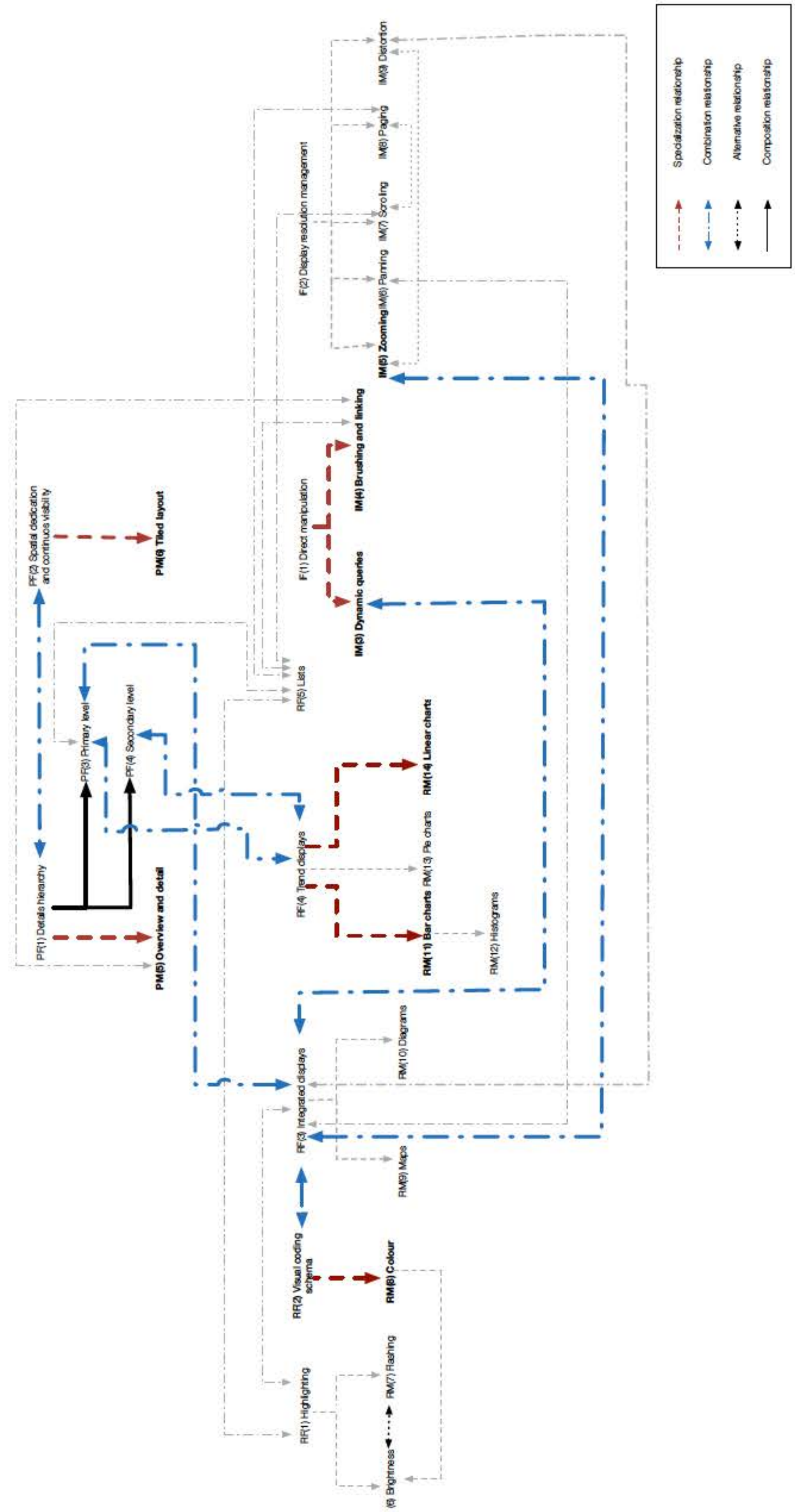


Fig. 5.2-8 Descriptive evaluation: Generated route by the refinement process in the second scenario

It is important to point out some of the main decisions made during this requirements refinement process. In order to address **RQ1**, *PF(1) Details hierarchy* was firstly specialized to *PM(6) Overview and detail* pattern. It also used *PF(2) Spatially dedicated, continuously visible displays*, which was in turn specialized to *PM(7) Tiled layout,* and two levels of detail of alarm information (*PF(3) Primary level of detail* and *PF(4) Secondary level of detail)*. In order to address **RQ2**, *RF(2) Visual coding schema* was specialized to *RM(5) Colour*. To address **RQ3**, *RF(3) Integrated displays* and *RF(4) Trend displays* were, in turn, replaced respectively by *RM(9) Maps, RM(11) Bar charts,* and *RM(14) Linear charts. RM(9) Maps* was also combined with *IM(5) Zooming* to allow a volunteer to obtain details of a selected portion of the map. To conclude, aiming at addressing **RQ4**, *IF(1) Direct manipulation* was specialized to both *IM(3) Dynamic queries* and *IM(4) Brushing and linking.* These patterns were also combined with *RM(9) Maps* and *RM(11) Bar charts.*

Next, **Step 4 (Organization of design patterns)** aims at organize the previous mapping according to the corresponding *purpose*. Accordingly, Table 5.2-8 lists the pair *requirement-design pattern* organized by *purpose*.

Presentation	
RQ1	PM(6) Overview and detail PM(7) Tiled layout
Representation	
RQ2 RQ3	RM(5) Colour RM(9) Maps RM(11) Bar charts RM(14) Linear charts
Interaction	
RQ3 RQ4	IM(3) Dynamic queries IM(4) Brushing and linking IM(5) Zooming

Table 5.2-8 Descriptive evaluation: Organization of design patterns by purpose in the second scenario

Fig. 5.2-9 Descriptive evaluation: Solution instantiation in the second scenario

Step 5 (Solution instantiation) is related to the definition of the final design solution dependent of the application domain. Accordingly, Fig. 5.2-9 shows how these design patterns were instantiated to develop the final design solution. Further details about this final design solution are described in [58].

5.2.4 Experimental Evaluation

The purpose of an experimental evaluation is to assess a designed artifact within controlled settings. In this work, this experimental evaluation assesses the design pattern language within controlled settings based on the use of the *controlled experiment* method. This method studies the designed artifact in order to determine a set of qualities. In this case, this evaluation attempts to assess the design pattern language's *usability* and *efficacy* by setting up an experiment with two rounds based on an assignment given to undergraduate students.

In the first round of the evaluation, two groups of participants, wherein one group used a paper version of the design pattern language and the other didn't, were requested to deliver an alarm visualization system sketch. This approach was used to measure differences in pattern language applicability based on the design material provided. In the second round of the evaluation, a new group of participants were requested to deliver an alarm visualization system sketch using an interactive version of the design pattern language. Taking into account the usability issues of the design pattern language reported by most participants in the first round of the evaluation, this approach was used to measure the usability of the design pattern language based on the use of an interactive graph-based version of the design pattern language. Further details about this evaluation are described in what follows.

First Round of the Evaluation

Experimental set-up

The first round of the evaluation was carried out within the context of a course titled "Designing Interactive Systems". The course is offered at the Spanish modality of the undergraduate program of the Computer Science and Engineering Department at the University Carlos III de Madrid, Spain. In total, 21 students attended to the course during the academic year 2014-2015. They on average had the same level of familiarity with design patterns based on the courses that the students had attended in previous semesters. They were given extra credit in coursework as an incentive to participate in this evaluation. All participants received the same preparatory material at the start of the evaluation. The design pattern language was delivered as oral introduction during a session of this course. The experiment was conducted during a second session by requesting the students an assignment on the design of an alarm visualization system paper sketch using a paper version of the design pattern language (see Fig. 5.2-10). The rationale for this was to try to avoid any electronic devices so that the tools themselves would not become a barrier to trying to sketch out ideas. They were divided into two groups, one group used the design pattern language and the other didn't. Within each group, students worked in an individual mode. From the 21 undergraduate students attending to the course on "Designing Interactive Systems", three students did not attend to the second session of the evaluation and did not deliver the assignment. Therefore, the total of participants in the experiment was 18, divided in groups of 9 participants each.

Fig. 5.2-10 First round of the Experimental evaluation: participants designing a sketch of the user interface of an alarm visualization system for emergency response

Procedure

The assignment requested to the students was framed within a crisis scenario for emergency response triggered by an earthquake. Earthquakes are one of the most challenging crisis situations that human operators within an emergency operation centre have to deal with. The description of this crisis scenario is provided in what follows:

> *"The 2011 Lorca earthquake was a moderate magnitude 5.1 earthquake that caused significant localized damage in the Region of Murcia, Spain. Centred at a very shallow depth of 1 km near the town of Lorca, a ninety-thousand-population city, it occurred on 11 May 2011, causing panic among locals and displacing many from their homes. A falling cornice killed three people. A total of nine deaths were confirmed, while dozens were reported injured. The situation was not re-established until three days later.*
>
> *From the 112 emergency operations centre of the Region of Murcia point of view, the earthquake started around 17:35 on 11 May 2011. During the next fifteen minutes, the alarm system registered a set of alarms from electromagnetics sensors with high priority and the calling centre received thousands of citizen callings related to seismic activity in the regions of Murcia, Albacete and Almería. This first activity was preceded by a magnitude 4.5 foreshock at 18:50, which resulted in an increasing registration of high-priority alarms located around the southeast of the Region of Murcia.*
>
> *Simultaneously, the population of Lorca started crowdsourcing about the developing situation, providing textual and graphical information through social networks. This kind of informal information helped human operators at the 112 emergency operations centre of the Region of Murcia to have a large picture of the situation: which areas were more affected, who needed helped, which kind of damages were produced"*

Based on this scenario, a set of design requirements was provided to the students (see Table 5.2-9). The deliverable of the assignment was a paper sketch of the user interface of an alarm visualization system for emergency response. This sketch should address the set of design requirements. It also should be documented by discussing design decisions made for each requirement as well as by describing the use of appropriate design patterns in the case of the group using them. The time given to finish the deliverable of the assignment was one hour and a half according to the schedule of a session of this course. One researcher directed this session and assisted the students throughout.

Design Requirements	
R1	The human operator should be made aware of new incoming alarms, which have not been accepted yet
R2	The human operator should be able to characterize the source of alarms, typology, and priority
R3	The human operator should understand the chronological order of alarms
R4	The human operator should be able to establish spatial relationships among alarms in order to identify the most damaged areas by the earthquake
R5	The human operator should be able to compare the distribution of alarms across sources, typologies, and priorities

Table 5.2-9 First round of the Experimental evaluation: Design requirements for the sketch

To conclude the experiment, a set of closed-ended questions (see Table 5.2-10) was asked to the students about the difficulties of the assignment and the usability of the design pattern language. They were divided into general-purpose questions and questions related to specific requirements of the alarm visualization system. As in the expert-based evaluation, a four-value Likert scale was used to collect the opinion of participants: *strongly agree* (4), *agree* (3), *disagree* (2), and *strongly disagree* (1). This scale was chosen to avoid neutral responses. For the group that used the design pattern language, these questions were extended with three open-ended questions (see Table 5.2-11) about the aspects of the design pattern language that resulted more difficult to them. This step typically took about 5 minutes, making the complete experiment ranges from one hour and thirty-five minutes to one hour and forty minutes.

Closed-ended questions		
General-purpose questions		
Q1	It was difficult to understand the requirements to address	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q2	It was difficult to find a visual solution to the requirement at hand	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q3	It was difficult to relate requirements among them	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q4	It was easy to find a visual solution to the requirement at hand	☐ 1 ☐ 2 ☐ 3 ☐ 4
Questions related to R1		
Q1	It was difficult to understand the requirements to address	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q2	It was difficult to find a visual solution to the requirement at hand	☐ 1 ☐ 2 ☐ 3 ☐ 4
Questions related to R2		
Q1	It was difficult to understand the requirements to address	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q2	It was difficult to find a visual solution to the requirement at hand	☐ 1 ☐ 2 ☐ 3 ☐ 4
Questions related to R3		
Q1	It was difficult to understand the requirements to address	☐ 1 ☐ 2 ☐ 3 ☐ 4

Closed-ended questions		
General-purpose questions		
Q2	It was difficult to find a visual solution to the requirement at hand	☐ 1 ☐ 2 ☐ 3 ☐ 4
Questions related to R4		
Q1	It was difficult to understand the requirements to address	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q2	It was difficult to find a visual solution to the requirement at hand	☐ 1 ☐ 2 ☐ 3 ☐ 4
Questions related to R5		
Q1	It was difficult to understand the requirements to address	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q2	It was difficult to find a visual solution to the requirement at hand	☐ 1 ☐ 2 ☐ 3 ☐ 4

Table 5.2-10 First round of the Experimental evaluation: Closed-ended questions about the difficulties of the assignment

Open-ended questions	
Q1	Which elements of the design pattern language caused more mental effort for you?
Q2	Was it difficult to identify the purpose of each design pattern?
Q3	Do you consider that the design pattern language allows you to work in an appropriate order or do you need to make decisions in advance? If yes, which kind of decisions are they?

Table 5.2-11 First round of the Experimental evaluation: Open-ended questions for the group of students using the design pattern language

Analysis and results

Two main types of data were collected in this evaluation: *paper sketches* of alarm visualization designs (see Annex B – Collection of sketches (First round of the Experimental Evaluation)), and *questionnaires*. Overall, students draw a total of 18 sketches, one from each participant in the form of A4 pages, and fulfilled 18 questionnaires. In order to analyse the usability of the design pattern language, three evaluation metrics were assessed, including, *task level of satisfaction*, *usage of design patterns*, and *usage of elements of the design pattern language*. Regarding the efficacy of the design pattern language, two evaluation metrics were evaluated, including, *completeness of generated designs* and *quality of generated designs*. Completeness of generated designs is defined as the coverage of generated designs of the set of design requirements. Quality of generated designs can be understood as the adherence of generated designs to a set of design principles for alarm visualization design.

Regarding to the participants' task level of satisfaction, Fig. 5.2-11 shows the mean of participants' responses to the questions with general purpose. It conveys that the group using the design pattern language had fewer problems to carry the assignment out. In particular, a strongly disagreement (coloured in red) with the question at hand, was obtained by Q3 *("It was difficult to relate requirements among them")* and the highest score, which reflects a strongly agreement (coloured in green) with the question at hand, was obtained by Q4 *("It was easy to find a visual solution to the requirement at hand").* However, Fig. 5.2-11 also shows that there were no big differences between groups in the understanding of the design requirements (Q1). These results suggest that the design pattern language can be characterized as useful to assist non-experienced designers to both relate design requirements and find appropriate visual solutions to the requirements at hand. However, they also suggest that the design pattern language doesn't provide a significant advantage over understanding design requirements without it, which it is highly dependent on both the level of abstraction of the design requirement and the designer's interpretation. As previous research on visualization design has already pointed out [91], high-level descriptions of design requirements are difficult to be input to the abstraction stage from the vocabulary of the specific domain into a more abstract and generic description that is in the vocabulary of visualization. It is for this abstraction process in which the proposed design pattern language seems to acquire more value for the participants.

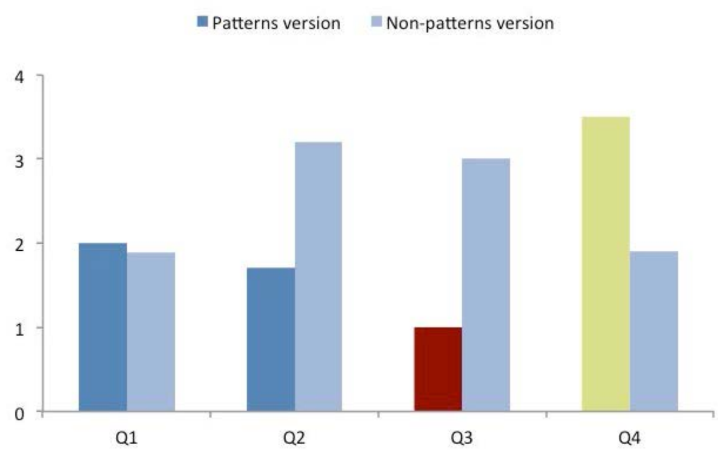


Fig. 5.2-11 First round of the Experimental evaluation: Results of participants. responses to the closed-ended questions with general purpose; for questions see Table 5.2-10

These general conclusions on the usability of the design pattern language were also supported by the results of the analysis of the participant's responses for each of the five design requirements to address in the sketch (see Fig. 5.2-12). These results shows that the group using the design pattern language found slightly more difficulties (coloured in red) than the group not using it to understand some specific design requirements such as DR2 *("The human operator should be able to characterize the source of alarms, typology, and priority")* or DR5 *("The human operator should be able to compare the distribution of alarms across sources, typologies, and priorities")*. However, it also displays that was easier for the group using the design pattern language to find visual solutions to the most design requirements.

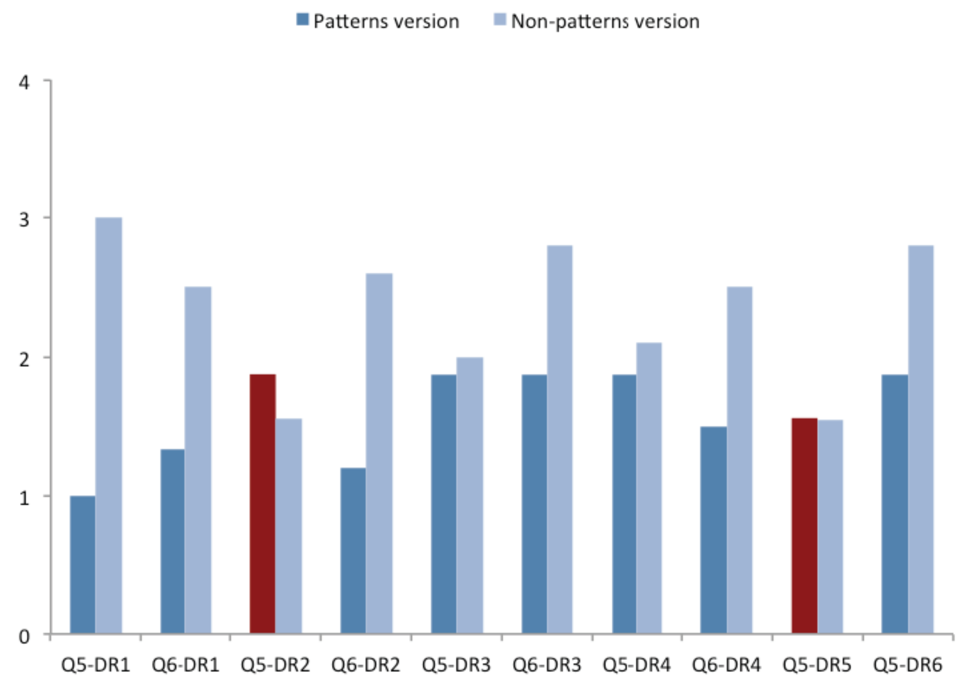


Fig. 5.2-12 First round of the Experimental evaluation: Results of participants' responses to the closed-ended questions with specific purpose; for questions see Table 5.2-10

Concerning to the usage of design patterns, the analysis of results followed the same strategy as in [36], assigning a value to each design pattern in the catalogue according to its suitability to the design problem being faced. In particular, four values were used: 0 if pattern should not be applied; 0,25 if its use is slightly recommended; 0,5 if is strongly recommended; and 1 if it is considered mandatory. Table 5.2-12 shows the list of design patterns used in this study.

<table>
<tr><th rowspan="2">Purpose</th><th colspan="2">Design Patterns</th></tr>
<tr><th>ID + name</th><th>Value</th></tr>
<tr><td>Presentation</td><td>PF(1) Details hierarchy
PF(2) Spatial dedication and continuous visibility
PF(3) Primary level of detail
PF(4) Secondary level of detail
PM(5) Overview and detail
PM(6) Tiled layout</td><td>1
0,5
1
0,5
1
1</td></tr>
<tr><td>Representation</td><td>RF(1) Highlighting
RF(2) Visual coding schema
RF(3) Integrated displays
RF(4) Trend displays
RF(5) Lists
RM(6) Brightness</td><td>1
1
1
1
1
1</td></tr>
</table>

Purpose	Design Patterns	
	ID + name	Value
	RM(7) Flashing RM(8) Colour RM(9) Maps RM(11) Bar charts RM(12) Histograms	0,5 0,5 1 1 0,25

Interaction	IF(1) Direct manipulation
	IF(2) Display resolution management
	IM(4) Brushing and linking
	IM(5) Zooming
	IM(7) Scrolling
	IM(8) Paging
	IM(9) Distortion
		1
		1
		1
		0,5
		0,5
		0,5
		0,5

Table 5.2-12 First round of the Experimental evaluation: Design patterns used in the evaluation. Design patterns with 0 value assigned are not displayed

Although it is not possible to determine why each pattern was frequently employed or not, in general, the trend shown in Fig. 5.2-13 implies the following issues. As in similar studies [36], the most commonly used design patterns were those that are more specific and detailed, classified under the *Mechanism* level. However, some counterexamples are *RF(1) Highlighting* and *RF(5) Lists,* feature-level design patterns that were used by most of participants. These design patterns are concerned with highlighting of information and using lists of items, which may be characterized as easy to comprehend and apply. Looking across the design patterns, it is also possible to see that presentation and representation patterns were most used; it may be that interaction patterns are difficult to implement and describe on a paper sketch. It is also worth noting that the only incorrect application of a design pattern occurred for a presentation pattern, *PM(6) Tiled layout*. It is possible to see that a tiled layout and a list are all mixed up on the sketch. It might be because they share a similar strategy of arranging items in a sequential way, being difficult to distinguish by a non-experienced designer.

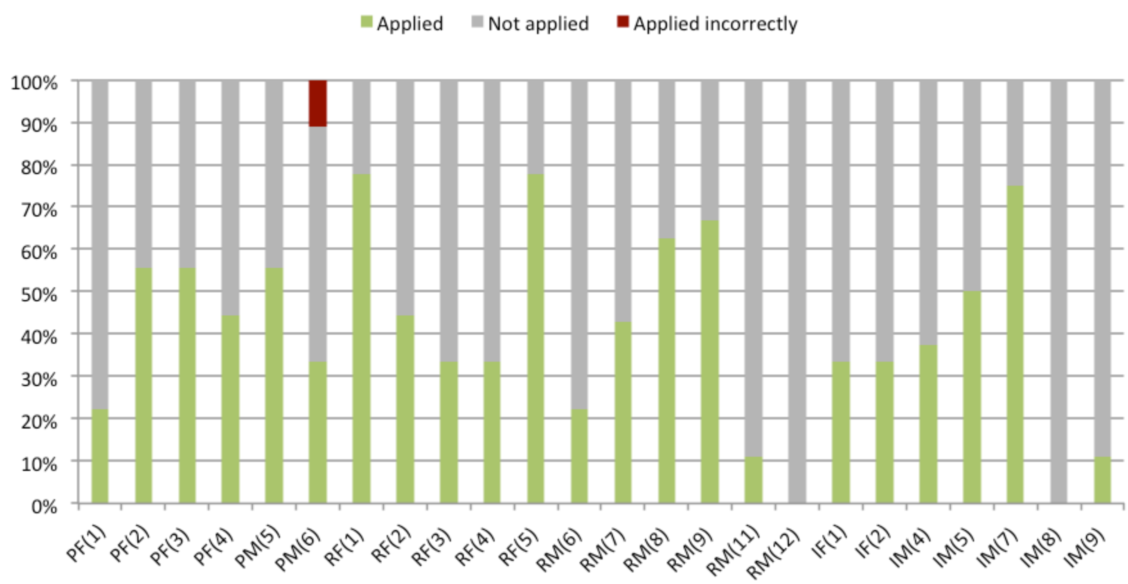


Fig. 5.2-13 First round of the Experimental evaluation: Results on the design patterns applied by the group of participants using the design pattern language; for patterns' names use Table 5.2-12

In order to know the participants' opinion about specific elements provided by the design pattern language, the participants' responses to the open-ended questions were analysed. In general, participants reported that they perceive as more difficult to understand the links between design patterns specified in the *"Relations"* block of each pattern, a result that has also been obtained in similar studies [36]. They needed to read a lot while starting to be aware about the design patterns and relate their mental design patterns to the patterns suggested by the language. As a positive feedback, it was obtained that they were satisfied with the resulting design and said that the design pattern language suggests strategies and considerations that otherwise they would not have included.

To conclude this evaluation, following a similar strategy to studies on the use of design patterns for other purposes [73,74,27], the judge of an researcher with experience on User Interface (UI) design evaluated, on a scale of three values ranging from *excellent* to *poor*, both *the adherence of the paper sketches to a set of recognized design principles* for alarm visualization design (see Table 5.2-13) and the number of design requirements (see Table 5.2-9) accomplished when using patterns versus non-pattern approaches. A three-value scale was used to collect the expert designer's scores per sketch: *excellent* (3), *acceptable* (2), and *poor* (1).

Design Principles for SA	
DP1	Organize information around the operator's goals
DP2	Support Level 2 SA (comprehension) information directly
DP3	Provide assistance for SA projections
DP4	Support global SA (an overview of the situation)
DP5	Support trade-offs between goal-driven and data-driven processing
DP6	Make critical cues for activation salient

Table 5.2-13 First round of the Experimental evaluation: Design principles defined by Endsley [44] used to evaluate the quality of designs

Regarding to the adherence of the sketches to this set of design principles, Fig. 5.2-14 shows that the sketches that had design patterns were rated as more than "acceptable" (2=acceptable) in 5 out of 6 design principles. However, although they were rated lower than the sketches that had not design patterns for DP1 *("Organize the information around goals")*, the higher standard deviation in this latter group suggests that this score was mostly obtained because of the innate creativity of some of the students. Moreover, the sketches that had design patterns yield more homogeneous scores and standard deviations across the six design principles. In this way, these results seem to indicate that the use of the design pattern language also adds some balance to the adherence to design principles across alarm visualization designs.

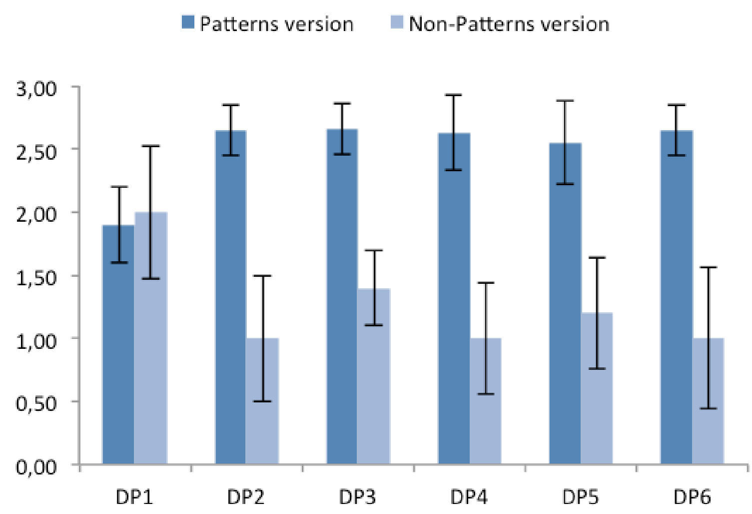


Fig. 5.2-14 First round of the Experimental evaluation: Results of the expert designer's ratings on the quality of the sketches on a scale 1-3 (3=excellent); for design principles use Table 5.2-13

Concerning to the coverage of design requirements when using patterns versus non-pattern approaches, Fig. 5.2-15 shows that the sketches that had design patterns were rated higher in 4 out of 5 design requirements. Particularly, it is important to highlight the good results obtained for requirements DR3 *("The human operator should understand the chronological order of alarms"),* DR4 *("The human operator should be able to establish spatial relationships among alarms in order to identify damaged areas by the earthquake")*, and DR5 *("The human operator should be able to compare the distribution of alarms across sources, typologies, and priorities").* Following the SA perspective proposed by Endsley, if a human operator is not able to comprehend the evolution of alarms over time, their spatial location, and their relevance, he/she won't be able to project ahead in order to avoid many undesirable situations. The ability to predict what relevant elements in the environment will do in the future allows human operators to be more proactive, and also very fast to respond when various critical events do occur. According to that, these results seem to suggest not only a better coverage of design requirements by the sketches that had design patterns but also a better coverage of relevant design requirements for supporting human operators' tasks.

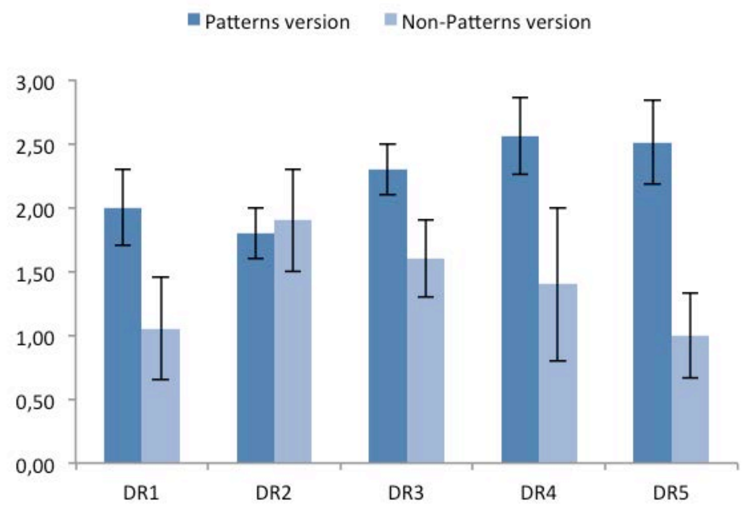


Fig. 5.2-15 First round of the Experimental evaluation: Results of the expert designer's ratings on the completeness of the sketches on a scale 1-3 (3=excellent); for design requirements use Table 5.2-9

Second Round of the Evaluation

This second round of the experimental evaluation was motivated by the usability issues reported by most of participants on the difficulties to understand the links between design patterns through the use of a paper version of the design pattern language. Accordingly, this second round was focused on measure the usability of the design pattern language based on the use of a different representation method. This second round of the evaluation tried to reproduce the evaluation settings of the first round by selecting participants with a similar profile and level of familiarity with design patterns. They were requested to carry out the same assignment than in the first round of the evaluation.

Experimental set-up

Volunteers were recruited via email and oral advertisement. A total of 9 participants took part in this second round of the evaluation. The participants were again undergraduate students enrolled in the undergraduate program of the Computer Science and Engineering Department at the University Carlos III de Madrid, Spain. All participants on average had the same level of familiarity with design patterns based on the courses that the students had attended in previous semesters. All participants received the same preparatory material at the beginning of the session. In particular, the design pattern language was delivered as an oral introduction during the first 15 minutes of the session.

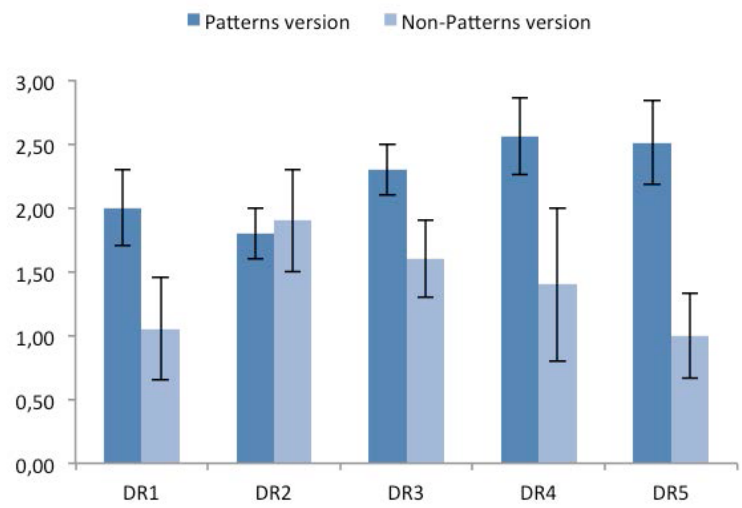


Fig. 5.2-15 First round of the Experimental evaluation: Results of the expert designer's ratings on the completeness of the sketches on a scale 1-3 (3=excellent); for design requirements use Table 5.2-9

SECOND ROUND OF THE EVALUATION

This second round of the experimental evaluation was motivated by the usability issues reported by most of participants on the difficulties to understand the links between design patterns through the use of a paper version of the design pattern language. Accordingly, this second round was focused on measure the usability of the design pattern language based on the use of a different representation method. This second round of the evaluation tried to reproduce the evaluation settings of the first round by selecting participants with a similar profile and level of familiarity with design patterns. They were requested to carry out the same assignment than in the first round of the evaluation.

Experimental set-up

Volunteers were recruited via email and oral advertisement. A total of 9 participants took part in this second round of the evaluation. The participants were again undergraduate students enrolled in the undergraduate program of the Computer Science and Engineering Department at the University Carlos III de Madrid, Spain. All participants on average had the same level of familiarity with design patterns based on the courses that the students had attended in previous semesters. All participants received the same preparatory material at the beginning of the session. In particular, the design pattern language was delivered as an oral introduction during the first 15 minutes of the session.

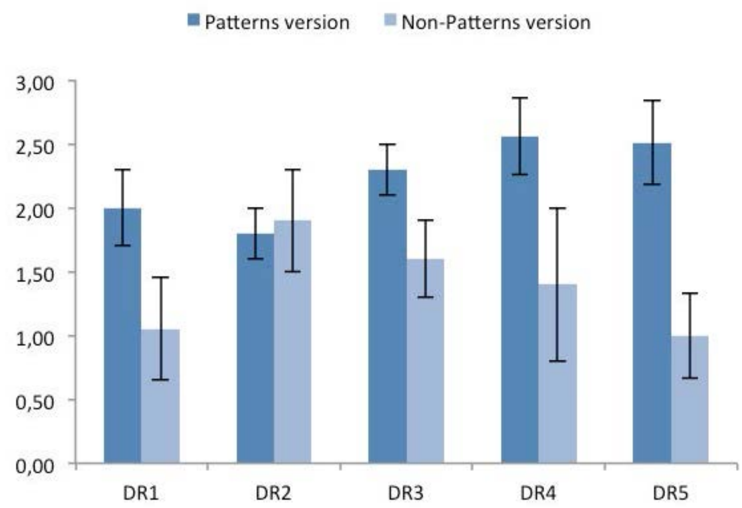


Fig. 5.2-15 First round of the Experimental evaluation: Results of the expert designer's ratings on the completeness of the sketches on a scale 1-3 (3=excellent); for design requirements use Table 5.2-9

SECOND ROUND OF THE EVALUATION

This second round of the experimental evaluation was motivated by the usability issues reported by most of participants on the difficulties to understand the links between design patterns through the use of a paper version of the design pattern language. Accordingly, this second round was focused on measure the usability of the design pattern language based on the use of a different representation method. This second round of the evaluation tried to reproduce the evaluation settings of the first round by selecting participants with a similar profile and level of familiarity with design patterns. They were requested to carry out the same assignment than in the first round of the evaluation.

Experimental set-up

Volunteers were recruited via email and oral advertisement. A total of 9 participants took part in this second round of the evaluation. The participants were again undergraduate students enrolled in the undergraduate program of the Computer Science and Engineering Department at the University Carlos III de Madrid, Spain. All participants on average had the same level of familiarity with design patterns based on the courses that the students had attended in previous semesters. All participants received the same preparatory material at the beginning of the session. In particular, the design pattern language was delivered as an oral introduction during the first 15 minutes of the session.

The experiment was conducted by requesting the students the same assignment provided to participants in the first round of the evaluation on the design of an alarm visualization system paper sketch. However, this time, participants were asked to interact with a graph-based version of the design pattern language (see Annex D - Interactive Version of the Design Pattern Language) implemented using the Vis JavaScript library [139]. In this graph-based version of the design pattern language, categories of design patterns are illustrated using colored labels and relations between design patterns are illustrated by edges with different formats pointing to the design patterns. Clickable labels allow users to visualize known uses of design patterns (see Fig. 5.2-16) and get access to detailed descriptions of design patterns. In addition, the graph-based version is also customizable and filterable. Users can customize the visualization by setting the *graph orientation*, *edges* types, and *separation between nodes*. A clickable legend allows users to click a legend entry to toggle design pattern categories (see Fig. 5.2-17). The first click highlights all the colored labels belonging to the selected category and fades out the rest of them. The second click toggles the colored labels for the rest of design patterns back to visible. Each participant worked in an individual mode.

Fig. 5.2-16 Second round of the Experimental evaluation: Using the graph-based version of the design pattern language to display known uses of a specific design pattern

Fig. 5.2-17 Second round of the Experimental evaluation: Using the graph-based version of the design pattern language to toggle design patterns categories

Procedure

Using the same scenario than in the first round of evaluation, participants were requested to deliver a paper sketch of the user interface of an alarm visualization system for emergency response. This sketch should address the same set of design requirements proposed in Table 5.2-9. It also should be documented by discussing design decisions made for each requirement as well as by describing the use of design patterns. As in the first round of the evaluation, the time given to finish the deliverable of the assignment was one hour and a half. One researcher directed this session and assisted the students throughout.

To conclude the experiment, a set of questions was asked to the students about the usability of the design pattern language. They were divided into closed-ended questions related to the ease of use and learnability of the design pattern language and closed-ended questions regarding to the elements of the language used to create a design solution (see Table 5.2-14). These closed-ended questions were complemented with two open-ended questions (see Table 5.2-15). For closed-ended questions, a four-value Likert scale was used to collect the opinion of participants: *strongly agree* (4), *agree* (3), *disagree* (2), and *strongly disagree* (1). This step typically took about 15 minutes, making the complete experiment ranges from one hour and forty-five minutes to two hours.

Closed-ended questions		
Ease of use and learnability		
Q1	It thought the design pattern language was easy to use	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q2	I would imagine it would be difficult by most people to learn to use this design pattern language	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q3	I found this design pattern language not very cumbersome to use	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q4	It was difficult to understand the requirements to address	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q5	I felt confident using the design pattern language	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q6	It was difficult to understand the requirements to address	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q7	I found this design pattern language not very cumbersome to use	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q8	It was difficult to find a visual solution to the requirement at hand	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q9	It was easy to find a visual solution to the requirement at hand	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q10	It was difficult to relate requirements to each other	☐ 1 ☐ 2 ☐ 3 ☐ 4
Usage of elements provided by the design pattern language		
Q1	Given the design scenario, the selection of design patterns was based on my intuition	☐ 1 ☐ 2 ☐ 3 ☐ 4

Q2	Given a design requirement, the first task I carried out was to look for a design solution	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q3	After selecting a design pattern, I followed the relationships between design patterns to find new design solutions	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q4	My first task was to find a suitable design pattern to the design requirement at hand	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q5	After selecting diverse design patterns, I followed the relationships between them in order to address the design requirements	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q6	Given the design scenario, the selection of design patterns was based on the relationships between patterns proposed by the pattern language	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q7	The design pattern language guided me to design the proposed solution	☐ 1 ☐ 2 ☐ 3 ☐ 4
Q8	After selecting a design pattern, looking for new design solutions was based on my intuition	☐ 1 ☐ 2 ☐ 3 ☐ 4

Table 5.2-14 Second round of the Experimental evaluation: Closed-ended questions about the usability of the design pattern language

Open-ended questions	
Q1	What kind of elements of the design pattern language helped you to design the proposed solution?
Q2	Provide an argumentation for the selection and usage of a design pattern applied

Table 5.2-15 Second round of the Experimental evaluation: Open-ended questions about the usability of the design pattern language

Analysis and results

Two main types of data were collected in this evaluation: *paper sketches* of alarm visualization designs (see Annex C – Collection of sketches (Second round of the Experimental Evaluation)), and *questionnaires*. Overall, students draw a total of 9 sketches, one from each participant in the form of A4 pages, and fulfilled 9 questionnaires. In order to analyse the usability of the design pattern language, four evaluation metrics were assessed, including, *ease of use ratings*, *ease of learnability ratings*, *usage of design patterns*, and *usage of elements of the design pattern language*.

Regarding to the ease of use and learnability of the design pattern language, the same procedure than in the System Usability Scale (SUS) tool [15] was followed in order to calculate the final score. In particular, for items 1,3,5,7,and 9 the contribution is the scale position minus 1. For items 2,4,6,8, and 10 is 4 minus the scale position. In order to obtain the overall usability score, it is required to multiply the sum of ten score contributions by 10. As far as the participants' opinion is concerned, the score of all of the participants (see Table 5.2-16) was higher than 60, with an average score of nearly 70 over 100.

Participant	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Score
1	3	3	4	1	3	3	4	1	4	1	65
2	4	2	3	3	4	1	3	2	4	1	67,5
3	3	2	3	2	5	1	3	2	4	1	65
4	2	4	2	3	4	2	3	4	3	3	75
5	3	2	4	2	4	1	5	2	3	2	70
6	3	1	4	2	4	1	3	2	2	2	60
7	3	4	3	3	4	2	4	2	2	4	77,5
8	3	2	4	4	3	1	4	2	3	3	72.5
9	4	3	2	3	4	2	4	2	3	1	67,5

Table 5.2-16 Second round of the Experimental evaluation: Results and final score of each participant

In particular, Table 5.2-16 displays that the best ratio was obtained by the question Q1 *("I would imagine it would be difficult by most people to learn to use this design pattern language")* and the worst ones by Q2 *("I would imagine it would be difficult by most people to learn to use this design pattern language")* and Q10 *("It was difficult to relate requirements to each other")*.

Concerning to the usage of elements of the design pattern language, Fig. 5.2-18 shows the mean of participants' responses to the closed-ended questions. It displays that participants found the design pattern language as useful to create the proposed solution to the given design scenario. In particular, the highest agreement (coloured in green) with the question at hand, was obtained by Q7 *("The design pattern language guided me to design the proposed solution")* and the lowest scores, which reflects a strongly disagreement with the question at hand, were obtained by both Q1 *("Given the design scenario, the selection of design patterns was based on my intuition")* and Q8 *("After selecting a design pattern, looking for new design solutions was based on my intuition").* These results suggest that the design pattern language addresses the lack of experience of non-experienced designers in alarm visualization design by both providing them design solutions to design problems they have been requested to face and guiding them to connected design ideas to the problems at hand.

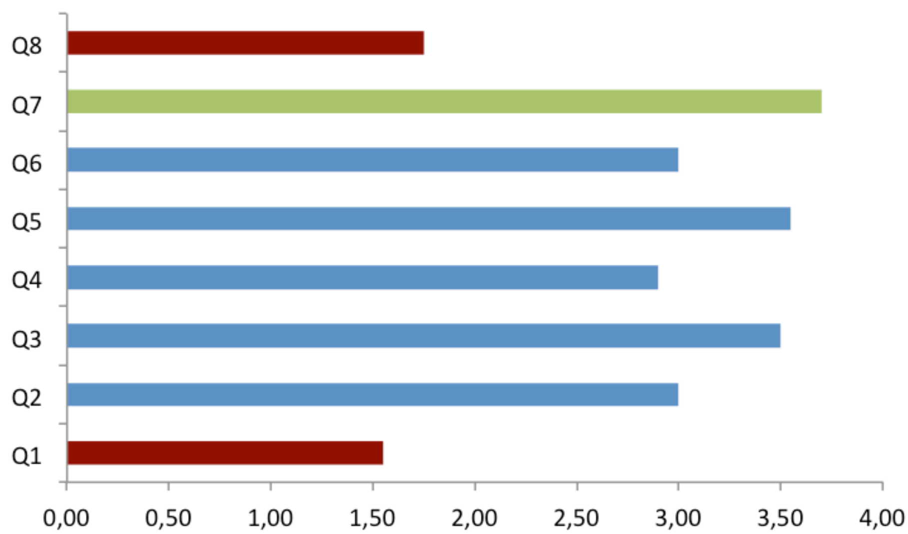


Fig. 5.2-18 Second round of the experimental evaluation: Results of the efficacy questions on a scale 1-4 (4=strongly agree); for questions use Table 5.2-14

To argument these promising conclusions about the usability of the design pattern language, the usage of design patterns in the sketches and the participants' responses to the open-ended questions displayed in Table 5.2-15 were analysed. Concerning to the usage of design patterns in the sketches, the analysis of the results followed the same strategy than in the first round of the evaluation. Although these results displayed in Fig. 5.2-19 are quite similar to those obtained in the first round of the evaluation (see Fig.

5.2-13), two main differences should be underlined. The first one is related to the absence of design patterns applied incorrectly. The second one is related to the increasing application of all strongly recommended design patterns such as the case of *IM(8) Paging*.

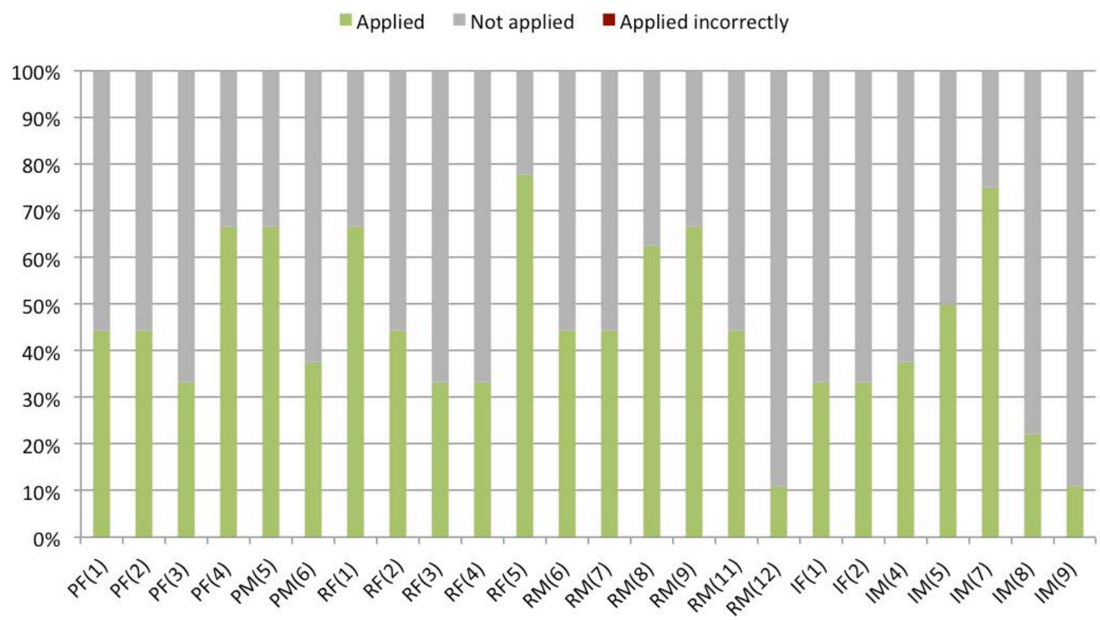


Fig. 5.2-19 Second round of the Experimental evaluation: Results on the design patterns used; for patterns' names use Table 5.2-12

The analysis of the results obtained from the participants' responses to the open-ended questions helped to clarify these two main differences. To these questions, most of participants not only reported about the utility of some elements of design patterns such as the design problems descriptions, the visual examples, and the relations between design patterns to create a design solution but also about the capabilities provided by the interactive version of the pattern language. In particular, all participants appreciated the interactive capabilities provided by such interactive version such as *filtering* and *hovering* on a specific design pattern in order to navigate more easily through the design pattern language. They found particularly useful the capability of quickly consulting visual examples of the proposed solution by hovering on each design pattern. In this way, these results seem to suggest that, in contrast to the previous experience of participants using a paper version of the pattern language, in this second round of the evaluation participants were able to spend more time understanding the descriptions of both the design problems and solutions proposed by design patterns instead of having to understand the structure

of the design pattern language. In other words, using this interactive graph-based version of the pattern language gave them more time to both understand how to apply design patterns in a correct way and considering the application of connected design patterns that otherwise they would have not considered.

Discussion

Generalizing from the two rounds of the evaluation, this experimental evaluation shows that this design pattern language can be characterized as useful to assist non-experienced designers to both relate design requirements and find appropriate visual solutions to the requirements at hand. It also proves that this design pattern language adds some balance to the adherence to design principles across alarm visualization designs. Similarly, it demonstrates that the use of the design pattern language allows not only a better coverage of design requirements by alarm visualization designs in general but also a better coverage of relevant design requirements for supporting human operators' tasks.

At the final point, this experimental evaluation expresses the usability limitations of the design pattern language related to the representation method. In order to be more usable, it suggests the use of an interactive graph-based version of the pattern language, which seems to both ease the navigation through the design pattern language and make more explicit the relations between design patterns provided by the pattern language.

Chapter 6. Conclusions

It's more fun to arrive a conclusion than to justify it

- ***Malcolm S. Forbes,*** **American publisher**

This research work has addressed the reuse of previous design knowledge by non-experienced designers for alarm visualization design. Designing alarm visualizations for operating control systems, as well as designing artifacts for other complex environments, requires the combination of design knowledge from different knowledge areas; in particular from Alarm Management, Human Factors, and Visualization Design. Since no single designer can be an expert in every relevant knowledge area, and becoming proficient may require years of experience, one relevant approach to assist in such design process is to reuse prior design knowledge. Reusing prior design knowledge can not only avoid repeated design effort but also help the designer adapt the original design to new situation for design innovation. However, existing design knowledge reuse approaches for alarm visualization design can be too abstract, not comprehensive enough, and loosely coupled, being difficult to be interpreted and applied by non-experienced designers. In keeping with that, this research work has presented a design pattern language to provide non-experienced designers with an easy access to the existing body of knowledge of alarm visualization design. The aim of a design pattern language is not to provide something that a designer must use; pattern languages are intended as tools that designers can use or not if they feel they have a better solution.

In particular, the design pattern language proposed in this research work encodes reusable alarm visualization design knowledge related to two fundamental and recurrent design challenges in alarm visualization design, *visual scalability* and *sense making*, in the form of 29 design patterns. It organizes such design patterns according to two different criteria validated by expert designers, *purpose* and *level of abstraction*, in order to support multiple ways of access depending on the design stage in which designers are.

Finally, in order to provide a cohesive organization of these design patterns for non-experienced designers, this design pattern language connects design patterns through five types of relations, including *generalization*, *specialization*, *alternative*, *composition*, and *combination*.

In this research work, the quality and utility of the proposed pattern language has been validated through four main evaluation methods including, respectively, an *analytical evaluation*, an *expert-based evaluation*, a *descriptive evaluation*, and an *experimental evaluation.* In particular, the experimental evaluation conducted with non-experienced designers has shown the capability of this design pattern language to add some balance to the adherence to alarm visualization design principles across designs. Similarly, it has also demonstrated the capability of this pattern language to assist non-experienced designers in the design of better alarm visualizations in terms of coverage of design requirements for supporting relevant human operators' tasks. However, such experimental evaluation has also uncovered some relevant considerations when displaying a design pattern language to non-experienced designers. In order to be more usable, it suggests the use of an interactive graph-based version of the pattern language, which seems to both ease the navigation through the design pattern language and make more explicit the relations between design patterns provided by the pattern language.

This final chapter begins with a summary of the major contributions that this research work makes to research. It continues with a description of other possible areas of application of the proposed solution. Finally, it concludes with a discussion of the limitations associated to this research work and with an exploration of future directions for the research.

6.1 Research Contributions

In order to illustrate the main contributions of this research work, this section makes use of the conceptual framework developed by Hevner et al. [59] for conducting design science research in the area of Information Systems (IS). This conceptual framework forms the basis for the methodological approach followed in this research work proposed by Offerman et al. [96]. Particularly, this conceptual framework is composed of three main different components: 1) the *environment* that defines the problem space in which reside the phenomena of interest. For IS research, it is composed of people, organizations, and their existing or planned technologies. In it are the goals, tasks, problems and opportunities that define business needs as they are perceived by people within the organization; 2) the *IS research* that is conducted. Given such problem space, IS research is conducted in two complementary phases, *building* and *evaluation* of artifacts designed to meet the identified need; and 3) the *knowledge base* that provides the raw materials from an through which IS research is accomplished. The knowledge base is composed of foundations and methodologies. According to this framework, the contributions of design science in IS research are assessed as they are applied to the business need in an appropriate environment and as they add to the content of the knowledge base for further research and practice. As Hevner et al. [59] states "*a justified theory that is not useful for the environment contributes as little to the IS literature as an artifact that solves a non-existent problem".* How this research work applies this conceptual framework is further described below and shown in Fig. 6.1-1.

Fig. 6.1-1 Illustration of the research contributions of this research work based on the conceptual framework proposed by Hevner et al. [59]

As shown in the *environment* component of the framework (see Fig. 6.1-1), the research conducted in this work mainly addresses a specific group of people: designers of alarm visualizations for operating control systems; in particular, non-experienced designers without professional or specialized knowledge in alarm visualization design. In order to design effective alarm visualizations for operating control systems, designers need to face two recurrent design challenges, *visual scalability* and *sense making*, which involve having design knowledge from different knowledge areas that may take years of experience to acquire. Moreover, existing design material for alarm visualization design can be too abstract, not comprehensive enough, and loosely coupled, being difficult to apply by non-experienced designers. The main contribution to the environment of this research work is a design artifact, a design pattern language, which non-experienced designers can use to reuse previous design knowledge on designing alarm visualizations **(Contribution 1).** This design pattern language not only contributes to provide useful starting points to recurrent design challenges for non-experienced designers but also may be used by more experienced designers of alarm visualizations to compare or explain new ideas, and to access to relevant alarm visualization design discussions.

As previously stated, the *main design artifact* built during *IS research* is the design pattern language itself, which can be used by non-experienced designers to reuse previous design knowledge for designing alarm visualizations. It has been analytically evaluated by examining the defined connections between design patterns. Similarly, it has been empirically evaluated in both expert-based evaluations and user studies. In addition, it has been applied to two different contexts and projects, including the design of alarm visualizations for the electric power grid operation and the design of alarm visualizations for emergency response.

Finally, regarding the *knowledge base* of this research work, it is possible to distinguish between foundations and methodologies. The *foundations* consists of descriptive models of alarm-initiated activities, Situation Awareness (SA), and Visualization Design that need to be carefully reviewed in order to characterize relevant factors affecting recurrent design challenges in alarm visualization design. Accordingly, a first contribution to the knowledge base of this research work comes from the results of such review. Such results extend the body of knowledge on designing alarm visualizations by framing the design space of alarm visualizations **(Contribution 2)**. Likewise, the concept of design knowledge reuse and existing design knowledge reuse approaches for alarm visualization design need to be reviewed in order to identify limitations and developing improvements. To build the proposed solution, a design pattern language, it is required to both have knowledge on *methodologies* on the construction of design pattern languages and review relevant sources for designing alarm visualizations such as reports on control systems, standards and design principles, and articles and books on visualization and interaction techniques that have been evaluated by several researchers or subject specialist in the academic community prior to accepting it for publication. Consequently, a second contribution of this research work to the knowledge base is the systematization of design knowledge on design challenges for alarm visualization design (**Contribution 3**). Similarly, the results of the analytical and empirical studies on the use of the design pattern language by non-experienced designers extend the body of knowledge in different areas, including the design knowledge reuse community and alarm visualization design. Moreover, this evaluation uncovers some considerations about the representation method of design pattern languages that may inspire future work on visualizing design pattern languages. To conduct these evaluations, it is required having knowledge on *methodologies* on evaluation methods for design knowledge reuse approaches and user studies.

According to this framework, the contributions of this research work are applied to an appropriate environment and they add to the content of the knowledge base for further research and practice.

6.2 Other Examples of Application

In order to illustrate other areas of application of the design pattern language, this section provides a description of its usage in heuristic evaluations. In keeping with the idea of using interaction design patterns in heuristic evaluation previously proposed by Botella et al. [13], other areas of application of this design pattern language may be specifically in *heuristic evaluations* on the *usability* of current alarm visualizations for operating control systems. In a heuristic evaluation, usability experts review the user interface of a specific system and compare it against accepted usability principles, so called the “heuristics” [92]. Additionally, designers can make use of these heuristics in order to assure to their applications the highest leve of usability as possible. However, the use of heuristics requires a rich expertise as the usability evaluation includes, not only negative aspects but also positive actions to improve the user interface. Accordingly, the idea described in this section is the use of the proposed design pattern language to help designers in the task of proposing improvements on current alarm visualizations for operating control systems. To illustrate this idea, in what follows, the application of this design pattern language to a specific usability issue detected in a real control system for operating the electric power grid is described. Such usability issue was detected during a heuristic evaluation carried out within the context of the Energos project. When such heuristic evaluation was conducted, the Nielsen's recommendations [92] were followed.

Fig. 6.2-1 shows the user interface of the control system with such usability issue. This usability issue was related to the visual display of alarms; in particular, to the display of the status, typology, and priority of upcoming alarms registered by the control system. Through the list of alarms shown in Fig. 6.2-1 it is difficult to visually distinguish the status, typology, and priority of such registered alarms.

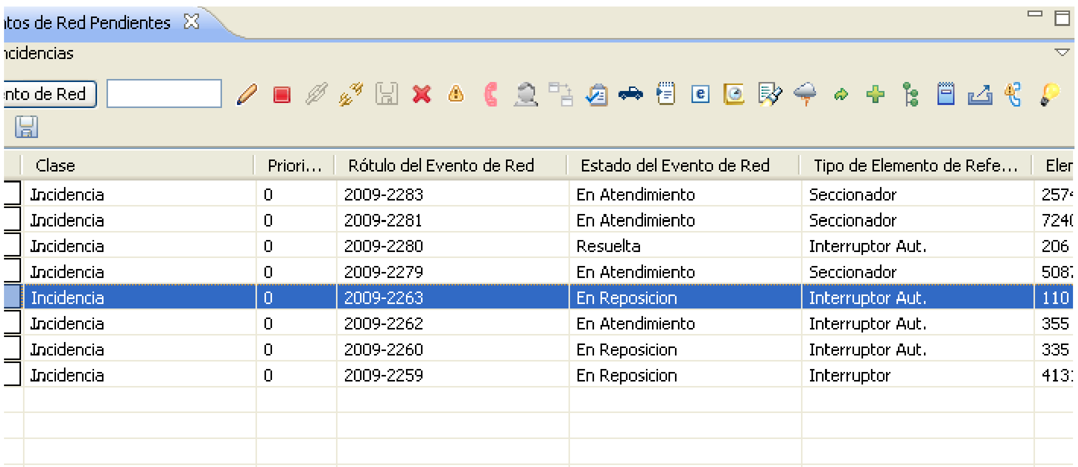
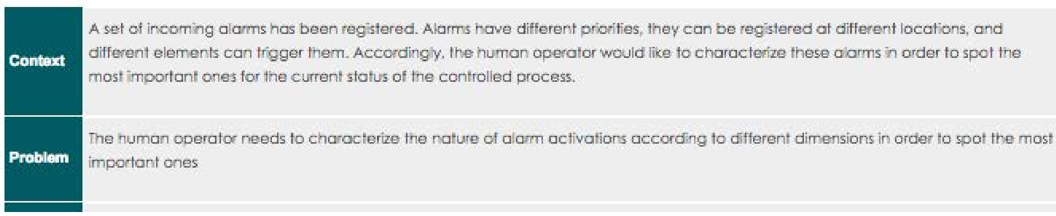


Fig. 6.2-1 List of alarms registered by the control system

In order to define how to solve this problem, the proposed solution is that the designer or usability expert uses the design pattern language and tries to find a design pattern or set of connected design patterns that can be used to fix the problem. In the case of this specific usability issue, adapting the process of use of the design pattern language defined in 4.3 Using the Design Pattern Language, the steps that a designer could follow in order to find an appropriate solution can be as follows:

- The designer explores the design pattern language and selects design patterns with *Representation* purpose. These patterns are referred to the assignment of specific visual characteristics to alarm data attributes in order to facilitate visual sense making by human operators.

- The designer takes a look on *Representation* patterns and according to the “Context” and “Problem” sections of each design pattern (see Fig. 6.2-2); he/she selects the *“Visual Coding Schema”* pattern. As shown in Fig. 6.2-3, this design pattern suggests the use of different visual properties to encode alarm information in a visual way in order evoke the human operator a ready understanding of alarm activations. However, in order to understand better how to implement such visual features, he/she follows the links established in the “Relations” section of the design pattern.



Context	A set of incoming alarms has been registered. Alarms have different priorities, they can be registered at different locations, and different elements can trigger them. Accordingly, the human operator would like to characterize these alarms in order to spot the most important ones for the current status of the controlled process.
Problem	The human operator needs to characterize the nature of alarm activations according to different dimensions in order to spot the most important ones

Fig. 6.2-2 Descriptions of the “Context” and the “Problem” solved by the Visual coding schema design pattern

Fig. 6.2-3 Descriptions of the “Solution” and “Known uses” of the Visual coding schema design pattern

- According to such relations provided by the design pattern language, the designer is able to propose the use of both the *"Colour"* pattern and the *"Flashing"* design pattern to support, respectively, the distinction of the typology and status of alarm information. In particular, the "Colour" pattern suggests the use of different colours to encode different types of alarms. The number of colours should be kept to seven or less, and be consistently applied. The "Flashing" pattern proposes to show new incoming alarms that require discrimination from others by using flashing. No more than two flash rates should be used. Finally, he/she also proposes the combination of such design patterns with the use of the *"Integrated displays"* pattern. Integrated displays are representations of the structure and relevant elements of a controlled process, in this case, the electric power grid. The set of alarms encoded using different colours and levels of brightness should be located close to the areas, components, or functions to which they are related. In this way, the human operator is assisted in the decision on the priority of these registered alarms. To see an example of the application of these design patterns see Fig. 6.2-4. It shows how some of these design patterns were instantiated to develop a final design solution in the context of the Energos project.

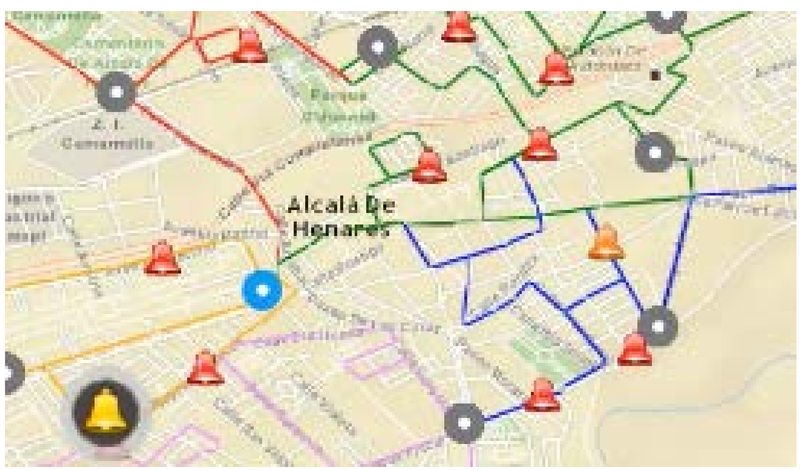


Fig. 6.2-4 Example of instantiation of combining "Colour", "Flashing", and "Integrated displays" patterns developed in the Energos project

6.3 Research Limitations and Future Research

As with any research, this research work has some limitations. Using this design pattern language to design alarm visualizations for operating control systems in other contexts and application domains may show both missing patterns and when descriptions of individual patterns are incomplete. These uses may in turn provide new interesting future research opportunities. In particular, one important future line of research can involve the investigation of how design patterns can help designers for different types of design situations and stages such as design evaluation. Like reusable components, design patterns develop in an iterative and evolving process through repeated reuse. Another related line of research involves investigating the appropriate diversity and redundancy of human operators' tasks for different application domains. In other words, this line of research would be concerned with the number of features that alarm visualization should offer.

Another possible area of research is conducting empirical studies to develop a more detailed understanding of the utility of design patterns. In a general sense, each design pattern can support a number of human operator's tasks; however, the knowledge of what specific conditions each design pattern supports particular tasks is still far from complete. Finally, another potential research topic is the development of interactive pattern programming environments and tools for the use of alarm visualization design patterns. In this sense, a computational representation of the proposed pattern language should be developed. To achieve this computational representation of the pattern language, an ontology-based approach might be a fitting approach. The semantic characteristics facilitated by ontologies enable the definition of models containing the pattern information along with the relations connecting the patterns. This approach would allow to computationally supporting the pattern selection process guiding designers in the design of alarm visualizations.

6.4 Related Publications

Parts of this research work have already been disseminated in the form of conference papers and journal papers. Such publications are listed in the following:

- Romero-Gómez, R. & Díaz, P. (2015). Towards a Design Pattern Language to Assist the Design of Alarm Visualizations for Operating

Control Systems. Paper presented at the 12th Conference of the Italian Chapter of AIS (ITAIS 2015)

- Herranz, S., Romero-Gómez, R., Díaz, P., & Onorati, T. (2014). Multi-view visualizations for emergency communities of volunteers. Journal of Visual Languages & Computing 11/2014 (Impact Factor: 0.56)
- Herranz, S., Romero-Gómez, R., & Díaz, P. (2014). Visualization techniques to empower communities of volunteers in emergency management. Proceedings of the Twentieth International Conference on Distributed Multimedia Systems, At Pittsburgh, Pensylvania DMS 2014.
- Romero-Gómez, R., Tena, S., Díez, D., & Díaz, P. (2013). The Application of Situation Awareness-Oriented Design to the Smart Grid Domain. Actas del XIV Congreso Internacional de Interacción Persona-Ordenador INTERACCION 2013.
- Romero-Gómez, R., Díez, D., & Aedo, I. (2013). Situation Awareness-oriented Alarm Visualizations: A next step in HSC environments. Paper presented at the Proceedings of the 4th International Conference on Information Visualization Theory and Applications IVAPP 2013

Annex A - The Catalogue of Design Patterns for Alarm Visualization Design

The following annex provides a detailed characterization of the **29 design patterns** that capture reusable design knowledge for alarm visualization design. A design pattern, in this research work, consists of eight descriptive blocks that provide textual information about several aspects of a pattern. Specifically, the description of every design pattern follows an adaptation of the format In a general sense, each design pattern can support a number of human operator's tasks; however, the knowledge of how an under proposed by Alexander et al. [6]. The meaning of each block is described as follows:

- ***Pattern identifier and name.*** An alphanumeric identifier is assigned to each design pattern. An order number in addition to both the first letter of its purpose category and the first letter of its scope category composes it. For example, a design pattern classified under the Presentation category with Feature level of abstraction is identified as *PF(X). Design Pattern name*. Afterwards, a pattern's name is also assigned to each pattern.
- ***Classification.*** The pattern's classification reflects the scheme introduced in Section 4.2.3 Defining a Classification Scheme.
- ***Context.*** This block describes the design context in which this pattern should be considered. Different solutions can arise from the same design problem occurring in different contexts. Accordingly, this block outlines the set of situations in which the pattern is effective in responding to what the human operator needs. It also describes any other design patterns that lead to this design pattern.
- ***Problem.*** This block provides a brief description of the design problem that trigger the usage of the pattern in question. It is an expression of what the human operator needs to do, perceive, or understand.

- ***Solution.*** This descriptive block outlines the crucial characteristics of the solution that address the problem identified. It captures details of how the solution can be executed, as well as further clarifying scope and context. Accompanying the textual information, each descriptive block features a graphic example that demonstrates its essential characteristics visually.
- ***Known uses.*** This block features two different graphic examples that demonstrate the essential characteristics of the design pattern visually*.*
- ***Rationale****.* This block delivers an argumentation for the usage of a pattern. This argumentation is mostly based on theoretical grounds related to *Alarm Management, Human Factors*, and *Visualization Design* that justify the use of the pattern.
- ***Relations****.* While most patterns can be used as individual entities to perform a certain task autonomously, the basic idea of a pattern language is to connect several modules with each other based on a concrete application context. Following the connective rules for pattern languages proposed by Salingaros [110], the collection of patterns presented in this work uses five types of relations: (1) a pattern is called the *specialization* of another pattern when it shares the same functionality but possesses more specialized characteristics or features; (2) the logical consequence of the specialization pattern is its inversion: the *generalization*. As the name implies, this pattern has rather generic features to serve a more universal purpose; (3) for some design purposes, there is more than one possible solution available. The *alternative* relation makes allowance for these use cases signalling the designer that he can choose from several options that lead, more or less, to the same result; (4) the *combination* relation is an alliance of patterns to be applied together in order to address a specific human operator task; and (5) the *composition* relation indicates that a pattern is composed of one or more other patterns, which can be characterized as *components*.

Despite the fact it has been tried to devote the same amount of space of each design pattern, because some patterns have been previously more developed than others, it provides more references for some design patterns than others.

PRESENTATION PATTERNS

PF(1) DETAILS HIERARCHY

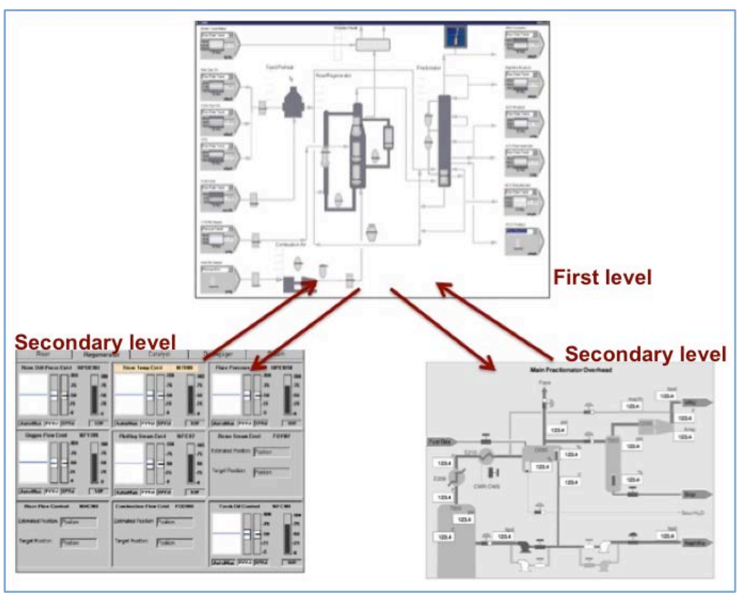
Classification. Purpose: Presentation. Level of abstraction: Feature.

Context. Within his area of responsibility and authority, the human operator needs to comprehend sets of incoming alarms displayed by alarm displays. Based on these displays, he would like to explore the overall alarm information space according to different alarm dimensions such as typology, priority, or location.

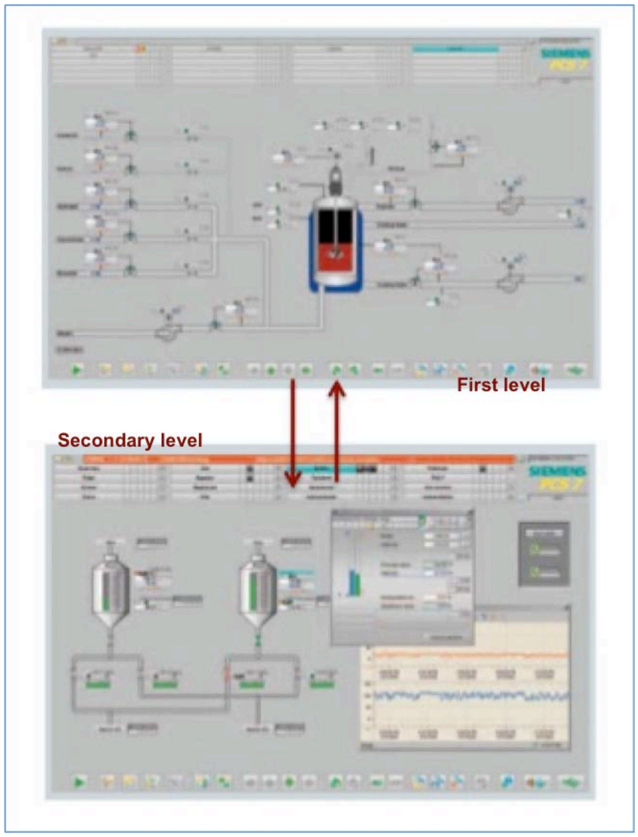
Problem. *The human operator needs to explore the overall alarm information space according to different alarm dimensions in order to diagnose the cause of a failure in the controlled process.*

Solution. Use a standard display hierarchy to present the multi-level views necessary for exploring alarm information. These levels follow an expected progression from the general to the more detailed in order to aid the operator in performing different tasks. A single first level can be associated with several secondary levels. Each secondary level can only be associated therefore with a single first level.

Known uses.



Annex Fig. A-1 An implementation of the Details hierarchy pattern: Hierarchy of levels applied to a control system interface for a power plant [107]



Annex Fig. A-2 An implementation of the Details Hierarchy pattern: Hierarchy of levels applied to the SIMATIC PCS 7 process control system developed by Siemens [121]

Rationale. A display hierarchy delivers a robust structure that encourages ready access to information while at the same time keeping important situation context and promoting efficient navigation to go deeper [107].

Relations.

- Composition relationship: *PF(3) Primary level of detail, PF(4) Secondary level of detail.*
- Specialization relationship: *PM(5) Overview and detail*

PF(2) Spatial dedication and continous visibility

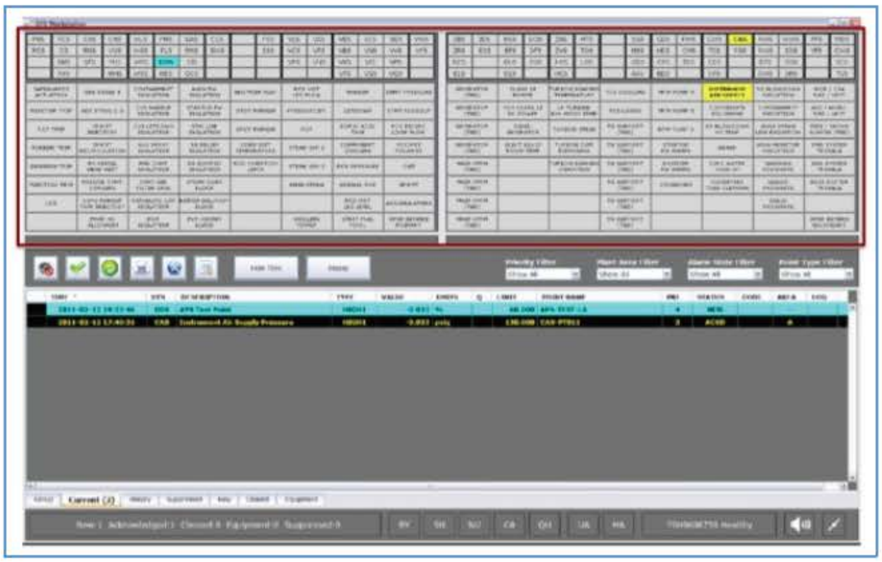
Classification. Purpose: Presentation. Level of abstraction: Feature.

Context. Under all controlled process conditions, even during minor disturbances, the human operator would like to know all key alarms that are directly safety related and important process alarms related to safety critical systems.

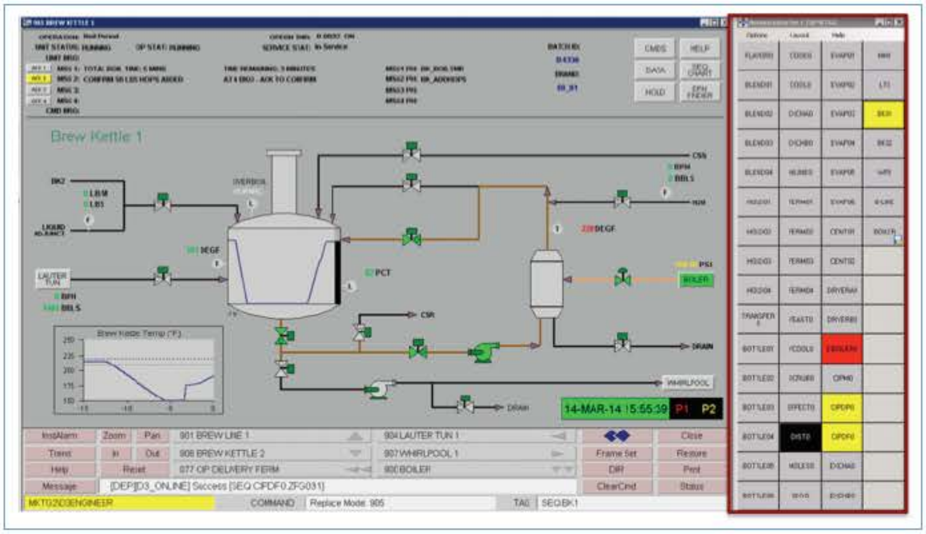
Problem. *The human operator needs to know at all times all key alarms in order to decide which alarms to deal with.*

Solution. Use spatially dedicated, continuously visible alarm displays to support response to all controlled process conditions. Spatial dedication means that the alarm messages always appear in the same position on the interface. Continuously visible means a parallel presentation method is used.

Known uses.



Annex Fig. A-3 An implementation of the Spatial dedication and continuous visibility pattern: The most relevant alarms (framed in red) are presented at the top of the Westinghouse alarm presentation system interface [143]



Annex Fig. A-4 An implementation of the Spatial dedication and continuous visibility pattern: The most relevant alarms (framed in red) are presented at the right side of the TotalVision Process Human Machine [62]

Rationale. A spatially dedicated, continuously visible display has generally been found during high-density alarm conditions to be superior to serial visual displays formats such alarm lists [44]. They ensure both an information rate and a presentation form that will remain manageable under all process conditions. They also allow the human operators a rapid detection and pattern recognition.

Relations.

- Generalization relationship*: PM(6) Tiled layout*

PF(3) Primary level of detail

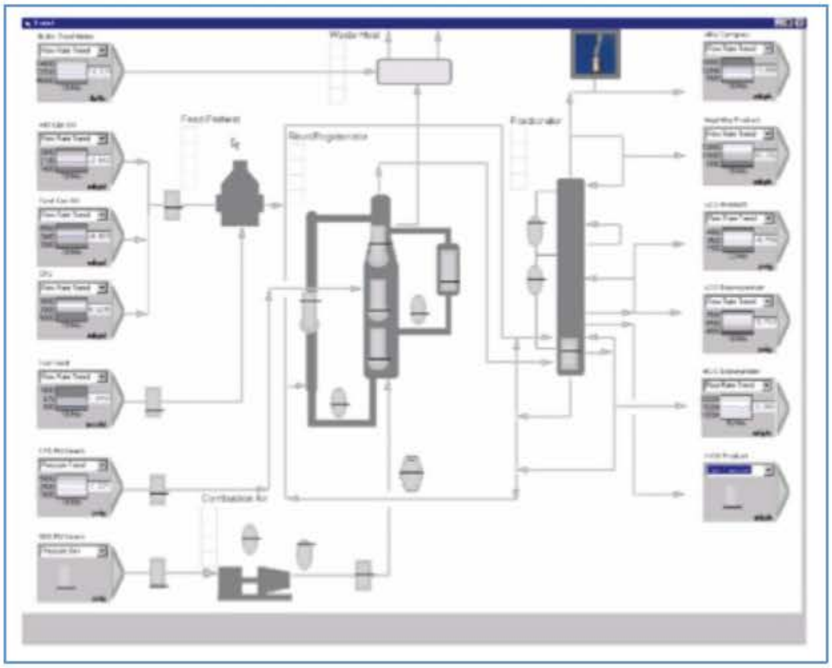
Classification. Purpose: Presentation. Level of abstraction: Feature.

Context. Within his area of responsibility and authority, the human operator needs to comprehend sets of incoming alarms displayed by alarm displays. In particular, the human operator would like to take a look into his entire operator's area of responsibility and know the status of the controlled process.

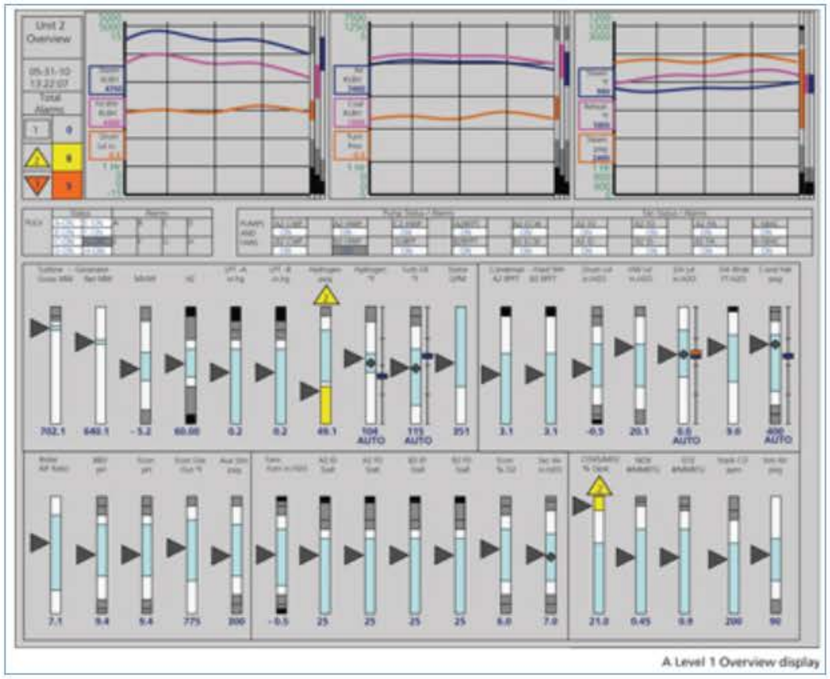
Problem. *The human operator needs to get an overview of the status of the controlled process control process in order to know whether this process is operating well.*

Solution. Use a first level of detail of alarm information that shows the broadest available view of the elements or spatial area under a single human operator's control. It is a big picture at-a-glance view of the status of the controlled process. It can incorporate a combination of separate alarm visual displays formats and integrated displays in which alarm information is integrated into the process displays. However, they are not necessarily a pictorial. They may be also a tabular listing of important status and summary information, with trends to help provide the big picture. All irrelevant information should be removed from this level to avoid information overload.

Known uses.



Annex Fig. A-5 An implementation of the Primary level of detail pattern: ASM-style primary level display [107].



Annex Fig. A-6 An implementation of the Primary level of detail pattern: ISA-style primary level display [64]

Rationale. A primary level of detail of alarm information allows communicating to the operator whether or not the controlled process is operating well [107].

Relations.

- Composition relationship (Component)*: PF(1) Details hierarchy*
- Combination relationship*: RF(3)Integrated displays, RF(4) Trend displays, RF(5) Lists*

PF(4) Secondary level of detail

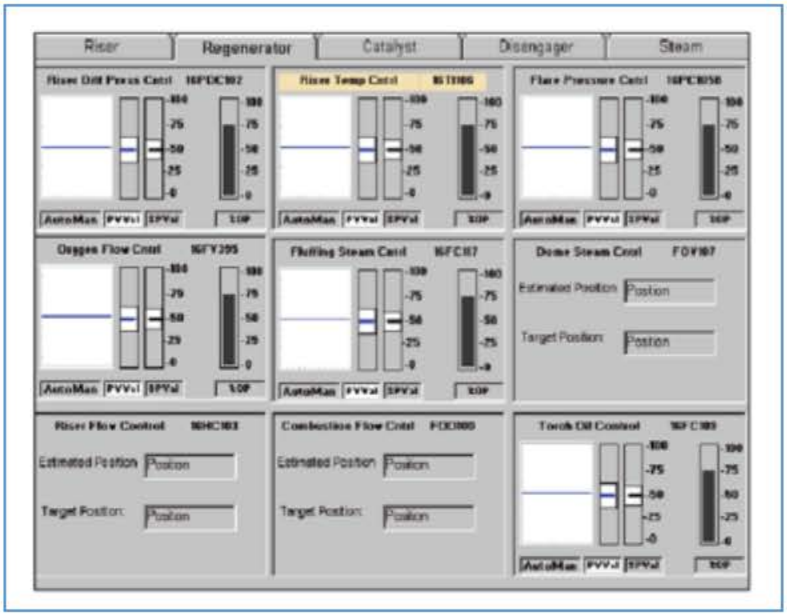
Classification. Purpose: Presentation. Level of abstraction: Feature.

Context. Within his area of responsibility and authority, the human operator needs to comprehend sets of incoming alarms displayed by alarm displays. After getting an overview of the status of the controlled process in his area of responsibility, the human operator would like to access to a more detailed information.

Problem. *The human operator needs to access to a more detailed information in order to better understand the status of the controlled process in his area of responsibility.*

Solution. Use secondary level of detail that presents all the alarm information regarding key elements of specific units or areas of the controlled process. This secondary level of detail allows operators to execute common and critical abnormal situation interventions. This level should be composed by a set of alarm selective displays, which show more detailed alarm information. It should also provide support displays, such as trend displays, diagnostics, calculation details, and so forth.

Known uses.



Annex Fig. A-7 An implementation of the Secondary level pattern: ASM-style secondary level display [107]

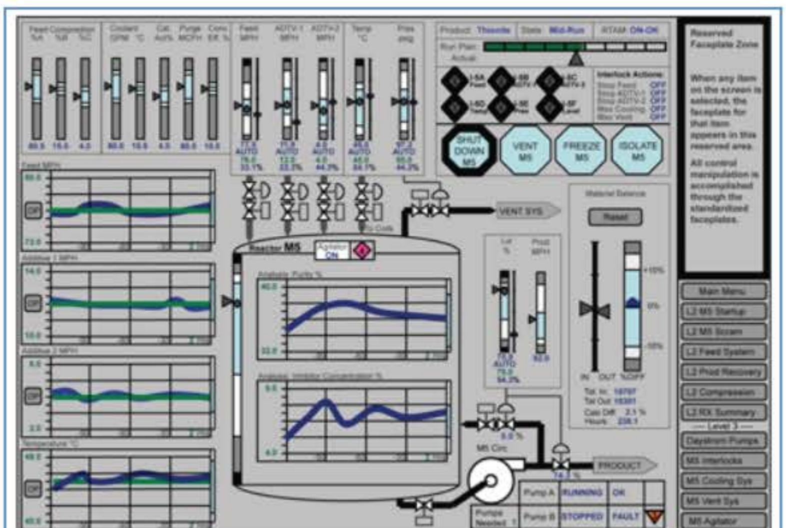


Figure 4: Example Level 2 display of a reactor

Annex Fig. A-8 An implementation of the Secondary level pattern: ISA-style secondary level display [64]

Rationale. A secondary level provides more detail to support what is already known or suspected from viewing the first level. If the overview looks normal, examining the secondary displays can be queried to verify this [107].

- **Relations.**
  - Composition relationship (Component)*: PF(1) Details hierarchy*
  - Combination relationship*: RF(4)Trend displays*

PM(5) Overview and detail

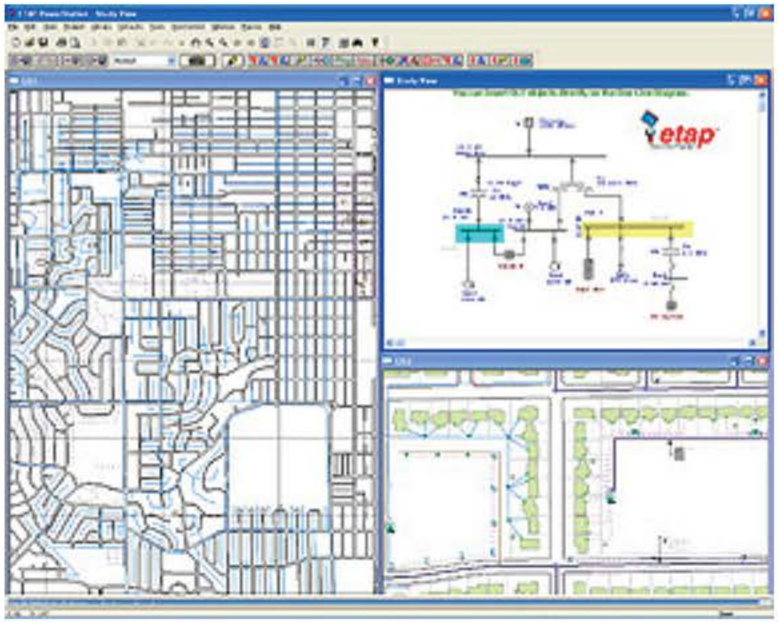
Classification. Purpose: Presentation. Level: Mechanism.

Context. Within his area of responsibility and authority, the human operator needs to comprehend sets of incoming alarms displayed by alarm displays. Based on these displays, he would like to explore the overall alarm information space according to different alarm dimensions such as typology, priority, or location.

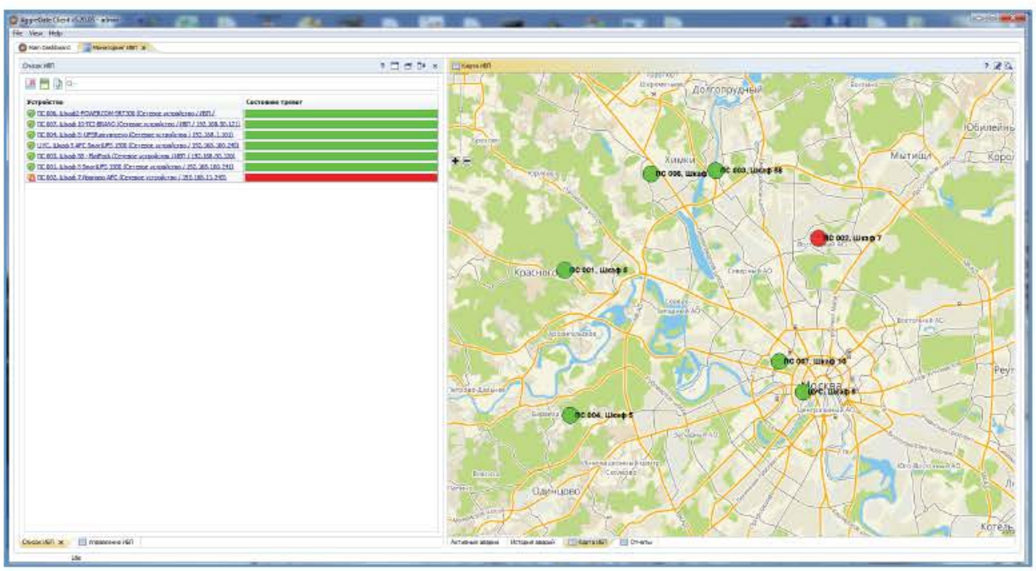
Problem. *The human operator needs to explore the overall alarm information space according to different alarm dimensions in order to diagnose the cause of a failure in the controlled process.*

Solution. Use a standard display hierarchy to present the multi-level views necessary for exploring alarm information. Display simultaneously the different levels of detail, each in a distinct presentation space. These different levels should be coordinated. It means that a selection of one alarm in a view should be highlighted in the other view to guide the operator to directly understand if the alarm is recurring or what the previous alarm from the same element was about. In any case, try to keep both levels visible for quick iteration.

Known uses.



Annex Fig. A-9 An implementation of the Overview and detail pattern: ETAP GIS Map provides a seamless view of the power system [123]



Annex Fig. A-10 An implementation of the Overview and detail pattern: AggreGate Network Manager is a monitoring tool for power supply systems. A geographical overview of a power grid infrastructure (right side) is displayed simultaneously with an alarm list (left side) [87]

Rationale. Overview and detail has been characterized as a highly effective method to deal with complexity [20]. It presents a high-level view of the situation and let the user drill down from that view into the details, as they need to, keeping both levels visible for quick iteration.

Relations.

- Specialization relationship: *PF(1) Details hierarchy*
- Combination relationship*: IM(4) Brushing and linking*

PM(6) TILED LAYOUT

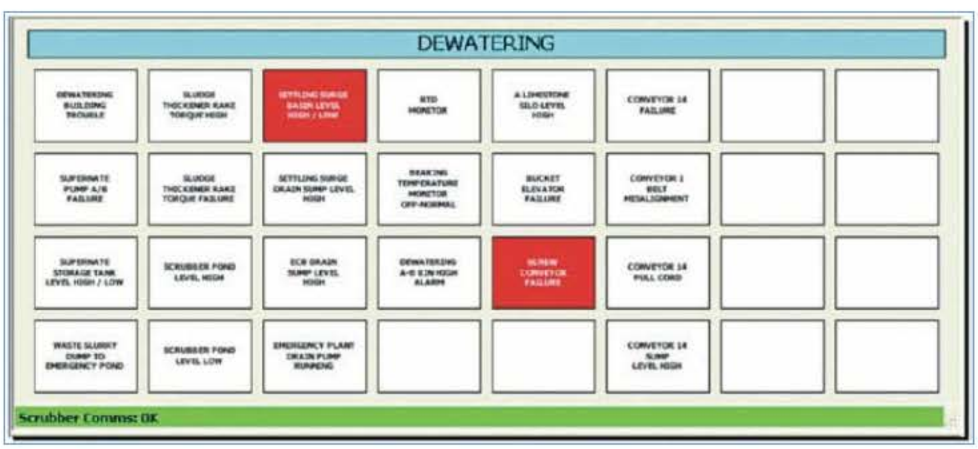
Classification. Purpose: Presentation. Level of abstraction: Mechanism.

Context. Under all controlled process conditions, even during minor disturbances, the human operator would like to know all key alarms that are directly safety related and important process alarms related to safety critical systems.

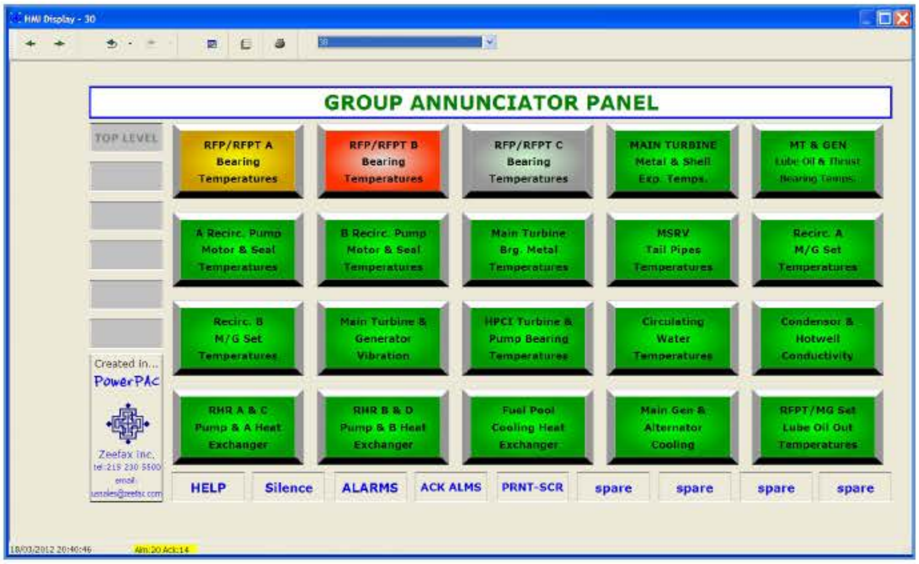
Problem. *The human operator needs to know at all times all key alarms in order to decide which alarms to deal with.*

Solution. Present key alarms in tiled layouts in a spatially dedicated position in which the same kind of alarm is always displayed in the same location. Arrange tiles in a grid, where all the tiles have the same width or splitters all with the same orientation separate same height and all the tiles in a group.

Known uses.



Annex Fig. A-11 An implementation of the Tiled layout pattern: Y-Plant Alert Alarm Annunciator System [146]



Annex Fig. A-12 An implementation of the Tiled layout pattern: Tiled display designed by Precision Digital Corporation (PDC) [99]

Rationale. Tiled layouts help operators to perceive alarms activity at a glance. As the same alarm is usually displayed in the same location, they also support human pattern matching [44]. In this way, over time, the operator may come to associate certain patterns of alarms with certain types of fault, thus frequent and familiar failures are likely to be readily recognizable.

Relations.

- Specialization relationship: *PF(2) Spatial dedication and continuous visibility*

REPRESENTATION PATTERNS

RF(1) HIGHLIGHTING

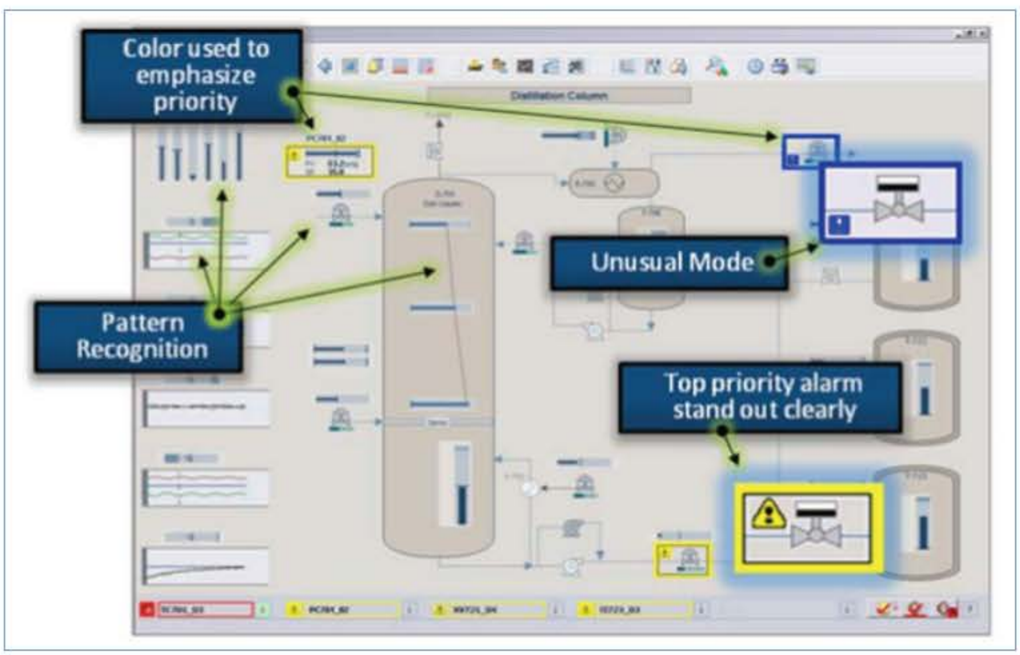
Classification. Purpose: Representation. Level of abstraction: Feature.

Context. A set of incoming alarms has been registered. The operator would like to both detect and distinguish them from alarms that have been accepted.

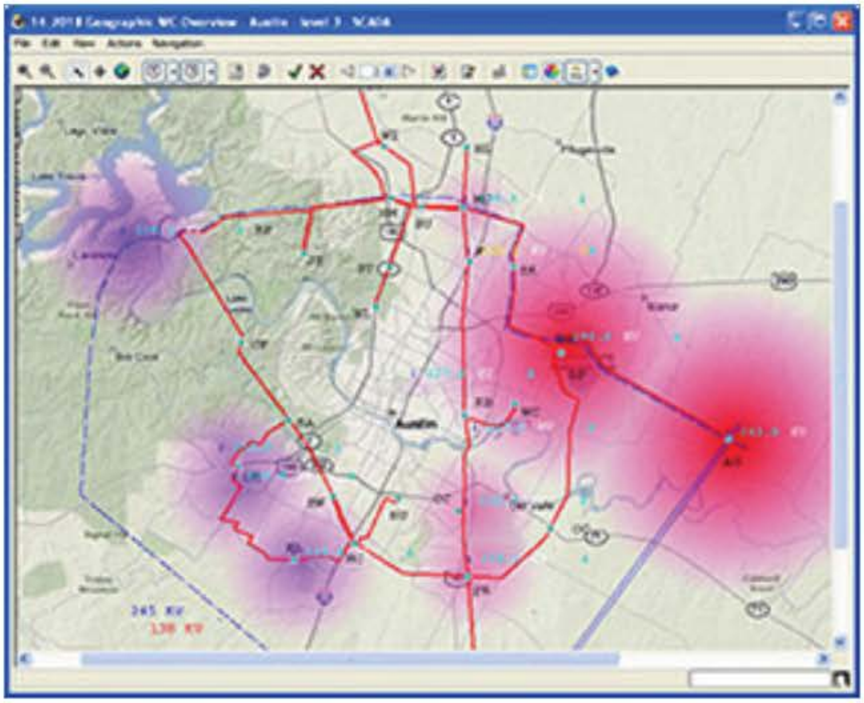
Problem. *The human operator needs to be made aware of new incoming alarms, which have not been accepted yet.*

Solution. Use highlighting mechanisms to direct the operator's attention to a set of incoming alarms. However, highlighting should be easily recognizable and minimized. A rule of thumb is to limit the maximum amount of highlighting to 10 per cent of the displayed alarm information. If a large proportion of the displayed alarms are highlighted, the highlighting will no longer be effective for directing the operator's attention. A particular highlighting mechanism should be used consistently. It means that highlighting mechanisms associated with critical conditions of the controlled process should not also be used in association with normal conditions.

Known uses.



Annex Fig. A-13 An implementation of the Highlighting pattern: DeltaV control system uses highlighting mechanisms such as bright colours to provide the operator an immediate access to alarms [43]



Annex Fig. A-14 An implementation of the Highlighting pattern: The PowerOn Reliance Energy Management System uses highlighting mechanisms such as bright colour contours to visualize grid areas in alarmed status [98]

Rationale. A major task of the visual system is to extract information about the great variation in illumination and viewing conditions [141].

Relations

- Generalization relationship*: RM(6) Brightness, RM(7) Flashing*
- Combination relationship: *RF(3) Integrated displays*

RF(2) Visual coding schema

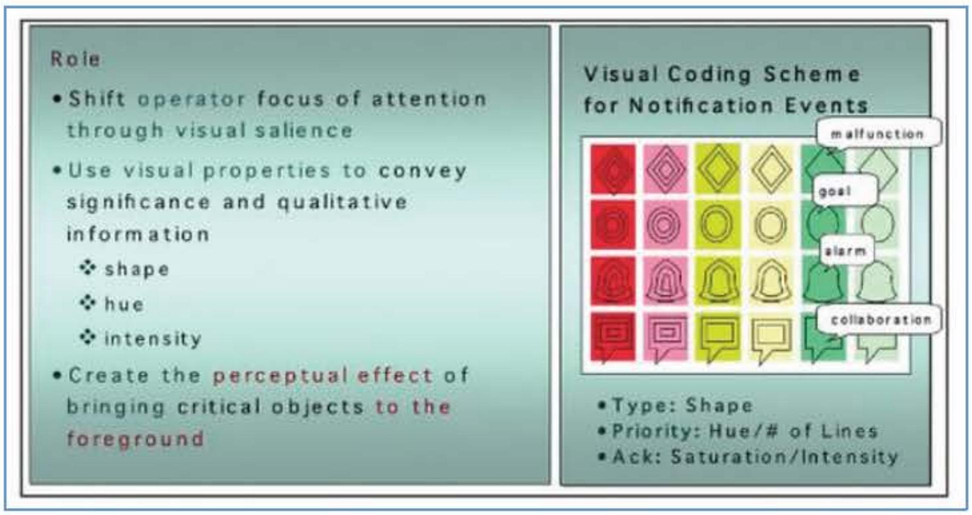
Classification. Purpose: Representation. Level of abstraction: Feature.

Context. A set of incoming alarms has been registered. Alarms have different priorities, they can be registered at different locations, and different elements can trigger them. Accordingly, the human operator would like to characterize these alarms in order to spot the most important ones for the current status of the controlled process.

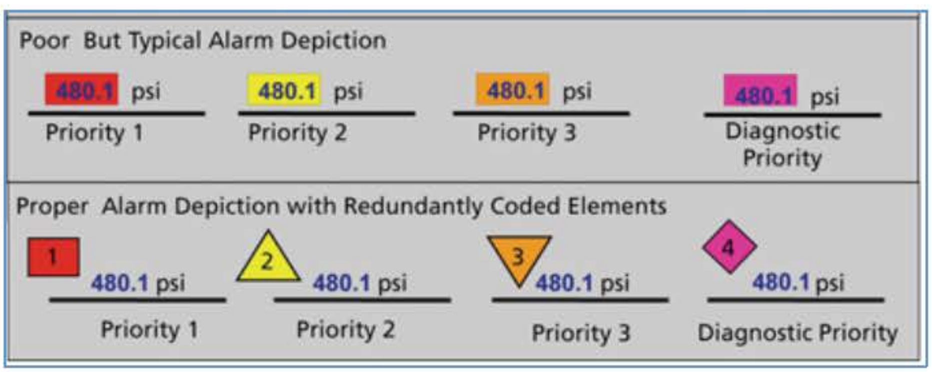
Problem. *The human operator needs to characterize the nature of alarm activations according to different dimensions in order to spot the most important ones.*

Solution. Use a visual coding schema composed by different marks and visual properties to evoke a ready understanding of the nature of alarm activations. Marks and visual properties should evoke a meaning and confirm that what is evoked is correct. Different marks and visual properties can be used to convey significance and reinforce their meaning. They should be employed conservatively and consistently.

Known uses.



Annex Fig. A-15 An implementation of the Visual distinction pattern: ASM-style visual coding schema [16]



Annex Fig. A-16 An implementation of the Visual distinction pattern: ISA-style visual coding schema [64]

Rationale. A visual coding schema speeds up understanding and minimizes confusion [16]

Relations

- Generalization relationship*: RM(8) Colour*
- Combination relationship: *RF(3) Integrated displays*

RF(3) Integrated displays

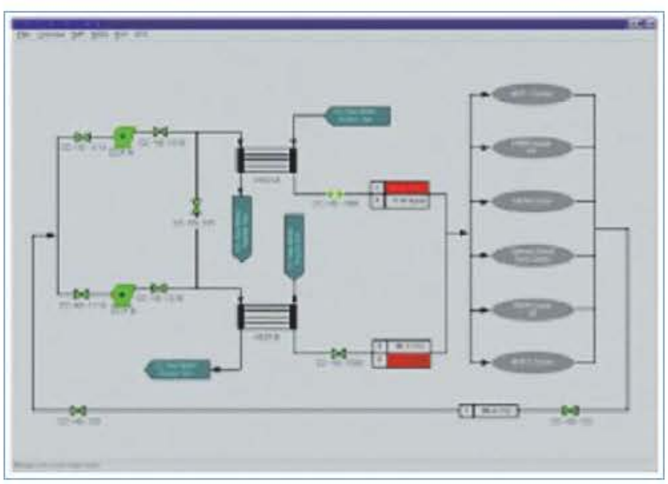
Classification. Purpose: Representation. Level of abstraction: Feature.

Context. The human operator would like to take a look into his entire operator's area of responsibility and know the status of the controlled process. He would like to comprehend the relationships between alarms and specific areas, components, systems, or functions of the controlled process.

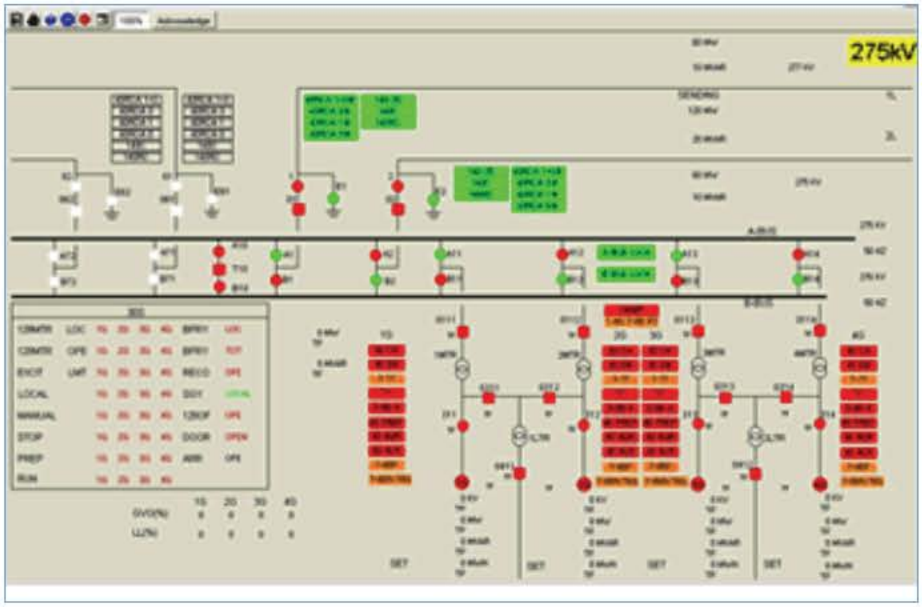
Problem. *The human operator needs to explore the relationships between alarms and relevant elements of the controlled process in order to establish the potential cause of a failure.*

Solution. Integrates alarm information in a consistent way across all process displays. Process displays represent the structure and relevant elements of a controlled process. It should be easy to see the nature of each alarm. Active main alarms should be easily distinguishable, salient features in the process displays. The process displays should show what alarms are defined and provide access to additional information on each alarm, such as alarm limit. When a process display exceeds one display page, display resolution mechanisms such as panning should be provided. The process displays should also allow the human operator to access to more detailed views of alarm information by using zooming and distortion mechanisms.

Known uses.



Annex Fig. A-17 An implementation of the Integrated displays pattern: An example of the integration of alarm information into a process display. In this type of display, the operator can access alarm information and process variables simultaneously [89]



Annex Fig. A-18 An implementation of the Integrated displays pattern: Toshiba's SCADA system provides one-line diagrams of power grids showing the alarm information and device statuses simultaneously [134]

Rationale. Combining both relevant process information and alarm information in the displays helps reduce the mental workload imposed on operators [260]

Relations

- Generalization relationship*: RM(9) Maps, RM(10) Diagrams*
- Combination relationship: *PF(3) Primary level of detail, RF(1) Highlighting, RF(2) Visual coding schema, IM(3) Dynamic queries, IM(5) Zooming, IM(6) Panning, IM(9) Distortion*

RF(4) Trend displays

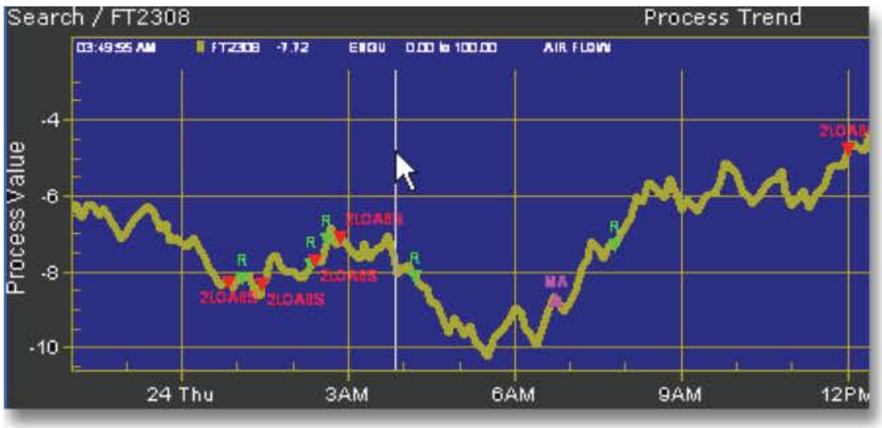
Classification. Purpose: Representation. Level of abstraction: Feature.

Context. A set of incoming alarms has been registered. The human operator would like to compare and understand the evolution of alarms across different dimensions such as priority, typology or time in order to avoid possible occurrences and deviations in the controlled process.

Problem. *The human operator needs to visualize the activity of alarms across different dimensions in order to project future states of the controlled process.*

Solution. Use trend displays to show the activity of alarms. Trend displays should be displayed with sufficient resolution in time and magnitude to ensure that rapidly changing variables can be observed and accurately interpreted. Trend displays should convey also enough information to allow the operator to interpret the data without referring to additional sources. They should form recognizable geometric patterns for specific abnormal conditions.

Known uses.



Annex Fig. A-19 An implementation of the Trend displays pattern: Trend display provided by AlarmAnalyst system [2]

Annex Fig. A-20 An implementation of the Trend displays pattern: DIASYS control system provides a trend display showing the operation status of facilities, and failure alarms in a power grid [34]

Rationale. Trend displays allow operators to make decisions about the performance of a variable or variables over time [44].

Relations

- Generalization relationship*: RM(11) Bar charts, RM(13) Pie charts, RM(14) Linear charts*
- Combination relationship: *PF(3) Primary level of detail, PF(4) Secondary level of detail*

RF(5) LISTS

Classification. Purpose: Representation. Level of abstraction: Feature.

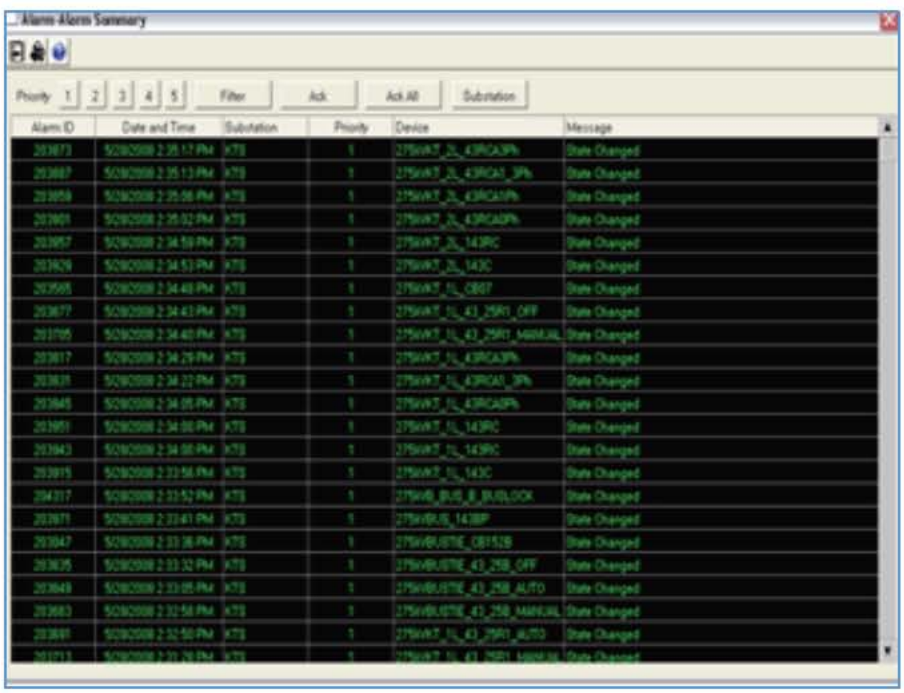
Context. A set of incoming alarms has been registered. The human operator would like to see the order in which alarms occurred, which is often useful for diagnosis of the underlying problem.

Problem. *The human operator needs to understand the chronological order of alarms.*

Solution. Show alarm information in lists and arranged it in a chronological order. A list is a display containing alphanumeric characters arranged by rows and columns. A list should be constructed so that row and column labels represent the information a user has prior to consulting the table. The left-most column should contain the labels for the row variables, and the top row should contain the labels for the column variables. When a list of alarms exceeds one display page, mechanisms to manage the display space should be provided.

Known uses.

Annex Fig. A-21 An implementation of the Lists pattern: Alarm lists displaying the chronological order of alarms in a CAS data logger interface [22]



Annex Fig. A-22 An implementation of the Lists pattern: Alarm summary in a list provided by the Toshiba's SCADA system [134]

Rationale. Lists have been characterized as useful in both retaining information about the order in which alarms occurred and displaying low-level detail alarm information [16].

Relations.

- Combination relationship: *PF(3) Primary level, IM(4) Brushing and linking, IM(7) Scrolling, IM(8) Paging*

RM(6) Brightness

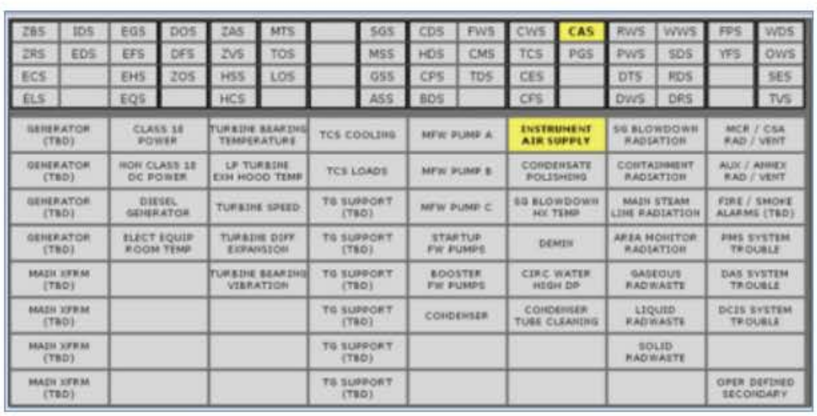
Classification. Purpose: Representation. Level of abstraction: Mechanism.

Context. A set of incoming alarms has been registered. The human operator would like to both detect and distinguish them from alarms that have been accepted.

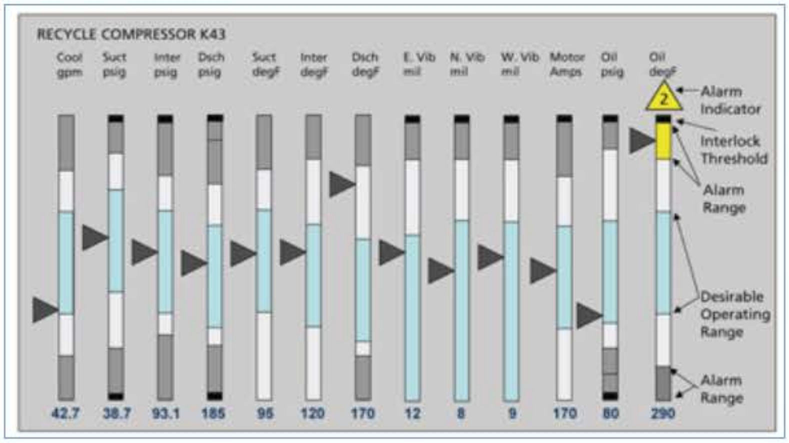
Problem. *The human operator needs to be made aware of new incoming alarms, which have not been accepted yet.*

Solution. Show new incoming alarms that require discrimination by using differences in colour brightness. Brightness coding should not be used in conjunction with shape or size coding. High brightness levels should be used to signify information of primary importance, and lower levels should be used to signify information of secondary interest.

Known uses.



Annex Fig. A-23 An implementation of the Brightness pattern: Use of brightness according to the priority of the alarm in the Westinghouse alarm presentation system interface [143]



Annex Fig. A-24 An implementation of the Brightness pattern: ISA recommends to depict the value's proximity to alarm ranges using colour brightness [64]

Rationale. Colour brightness allows highlighting properties in data [267].

Relations.

- Specialization relationship: *RF(1) Highlighting, RM(8) Colour*
- Alternative relationship: *RM(7) Flashing*

RM(7) Flashing

Classification. Purpose: Representation. Level of abstraction: Mechanism.

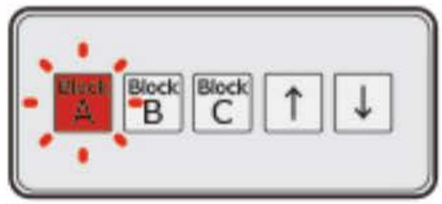
Context. A set of incoming alarms has been registered. The human operator would like to both detect and distinguish them from alarms that have been accepted.

Problem. *The human operator needs to be made aware of new incoming alarms, which have not been accepted yet.*

Solution. Show new incoming alarms by using flashing. An 'off' condition should never be used to attract attention to an alarm message. Flashing should not be used as a means to highlight routine information. It should only be used as an alerting code. When an operator must read a displayed item that is flash coded, an extra symbol such as an asterisk or arrow to mark the item should be used, and the marker symbol should flash rather than the item itself. No more than two flash rates should be used.

Known uses.

Annex Fig. A-25 An implementation of the Flashing pattern: Top priority alarm stand out by using flashing in DeltaV control system [43]



Annex Fig. A-26 An implementation of the Flashing pattern: Use of flashing to visualize the status change of an alarm in the Proface control systems [102]

Rationale. Flash coding generally reduces search times, especially in dense displays [37].

Relations.

- Specialization relationship: *RF(1) Highlighting*
- Alternative relationship: *RM(7) Brightness*

RM(8) Colour

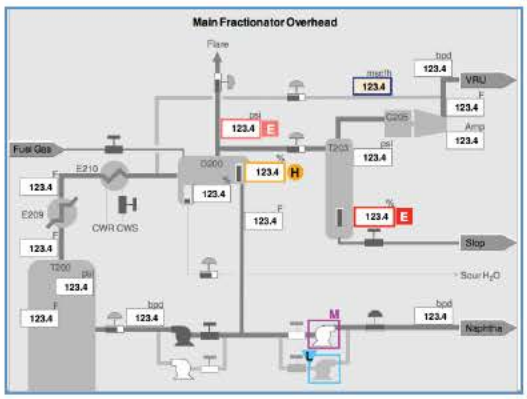
Classification. Purpose: Representation. Level of abstraction: Mechanism.

Context. A set of incoming alarms has been registered. Alarms have different priorities, they can be registered at different locations, and different elements can trigger them. Accordingly, the human operator would like to characterize these alarms in order to spot the most important ones for the current status of the controlled process.

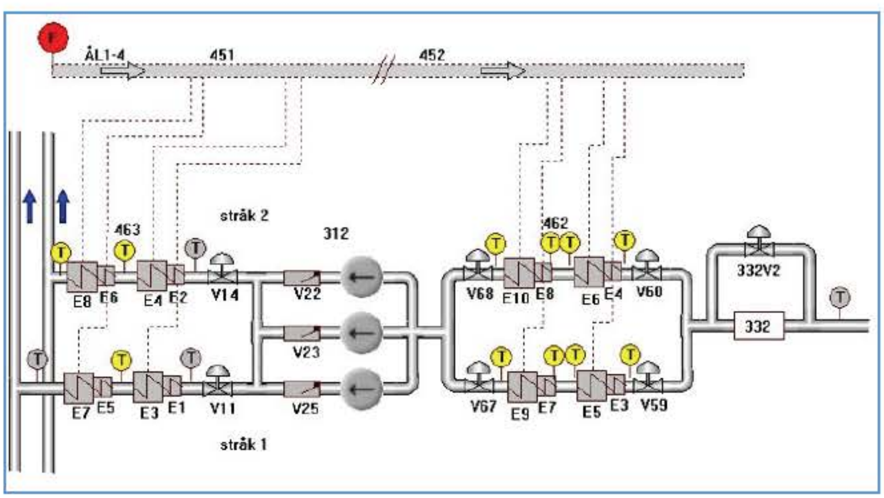
Problem. *The human operator needs to characterize the nature of alarm activations according to different dimensions in order to spot the most important ones.*

Solution. Code categories of alarms by using colours. The number of colours should be kept to seven or less, and be consistently applied. Seven corresponds to the number of items that can be generally being kept in short term memory. Once colours are assigned a specific use or meaning, no other colour should be used for the same purpose. The seven colour codes do not limit the coding of other information separate from the category information. When an operator must distinguish rapidly among several discrete categories of data, a unique colour should be used to display the data in each category. When the relative rather than the absolute values of a variable are important, gradual colour changes, as a tonal code, should be used to show the relative values of a single variable. Finally, brighter and/or more saturated colours should be used when it is necessary to draw a user's attention to critical data.

Known uses.



Annex Fig. A-27 An implementation of the Colour pattern: Light blue for low-priority alarms, orange-yellow for high-priority alarms, red for emergency-priority alarms (and pale red for acknowledged emergency-priority alarms, magenta for off-normal operating conditions) [16]



Annex Fig. A-28 An implementation of the Colour pattern: Yellow for high-priority alarms and red for emergency-priority alarms [5]

Rationale. Coding based on the use of colour is an important means for representing categorical information in displays [124].

Relations.

- Generalization relationship: *RM(6) Brightness*
- Specialization relationship: *RF(2) Visual coding schema*

RM(9) Maps

Classification. Purpose: Representation. Level of abstraction: Mechanism.

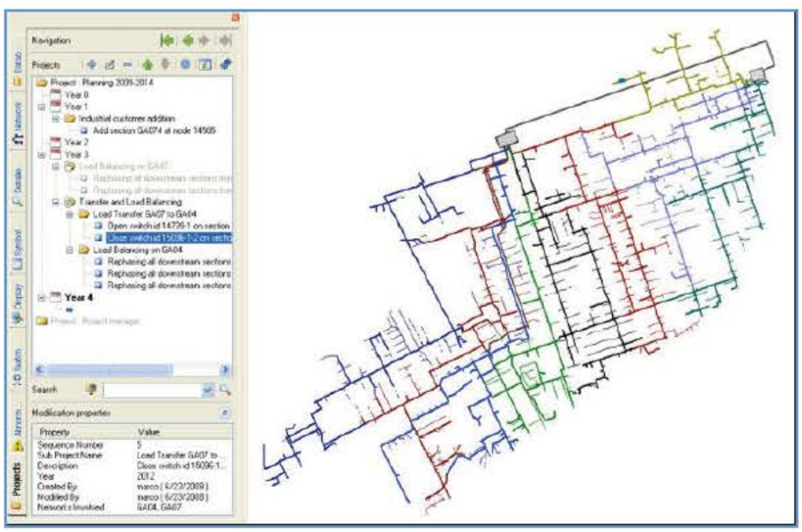
Context. A set of incoming alarms has been registered. The human operator would like to comprehend the spatial relationships between alarms and specific elements of the controlled process.

Problem. *The human operator needs to explore the spatial relationships between alarms and specific elements of the controlled process in order to establish the potential cause of a failure.*

Solution. Position alarms consistently in relation to spatial locations of the controlled process by using maps. A map is a visual display format medium that represents a spatial location and maps relevant elements of the controlled process to their geographical location. The choice about the detailed graphic specifications, such as size and resolution of the depicted area, level of detail, or the types of alarms included, varies depending on the specific application scenario. Significant features of a map should be labelled directly on the display unless cluttering or obscuring of other information would result. As a practical matter, map displays can get very crowded. Under these circumstances, some other approach to map labelling should be considered to avoid crowding.

Known uses.

Annex Fig. A-29 An implementation of the Maps pattern: ClearView SCADA Geographical Information System enables visualization of alarm information organized geographically with drill down and display capability into the assets to obtain detailed information about the alarm or event [29]



Annex Fig. A-30 An implementation of the Maps pattern: Geographic map of an electrical distribution network provided by the CYMDIST distribution analysis software [30]

Rationale. Maps represent the fundamental mode of display for information that have spatial attributes and can be immediately translated into real-world situations.

Relations.

- Specialization relationship: *RF(3) Integrated displays*

RM(10) Diagrams

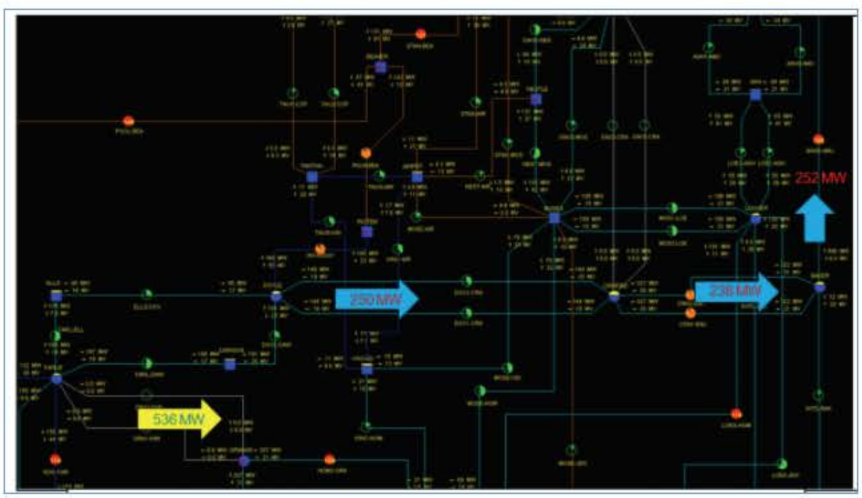
Classification. Purpose: Representation. Level of abstraction: Mechanism.

Context. A set of incoming alarms has been registered. The human operator would like to comprehend the functional relationships between alarms and specific elements of the controlled process.

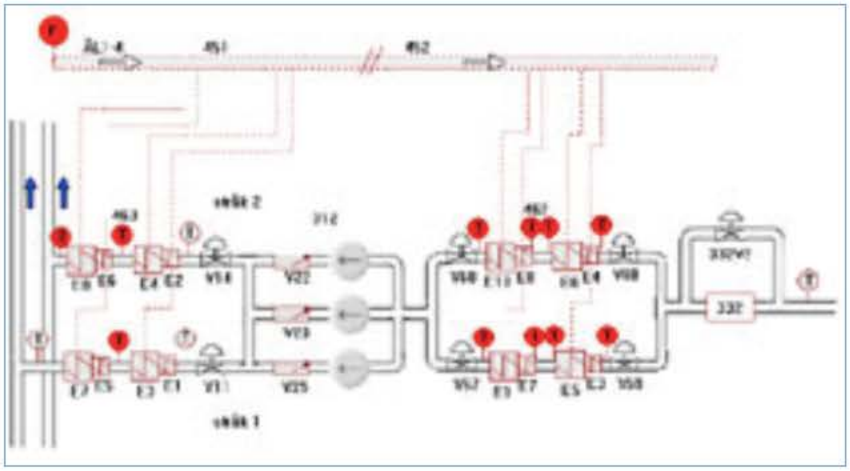
Problem. *The human operator needs to explore the functional relationships between alarms and specific elements of the controlled process in order to establish the potential cause of a failure.*

Solution. Map consistently the relationship between alarms and specific elements of the controlled process by using diagrams. A diagram is a visual display format that combines graphics and alphanumerics to reflect component relationships. They should contain the minimum amount of detail required to yield a meaningful pictorial representation. Controlled process components represented on lines should be identified. All flow path line origin and destination points should be identified. Flow directions should be clearly indicated by distinctive arrowheads.

Known uses.



Annex Fig. A-31 An implementation of the Diagrams pattern: Diagram of the PNNL Electricity Infrastructure [55]



Annex Fig. A-32 An implementation of the Diagrams pattern: Diagram of different elements of a nuclear power plant provided by the GoalArt control system [5]

Rationale. Diagrams have the advantage of providing a more direct spatial mapping between the control room and the controlled process [44].

Relations.

- Specialization relationship: *RF(3) Integrated displays*

RM(11) Bar charts

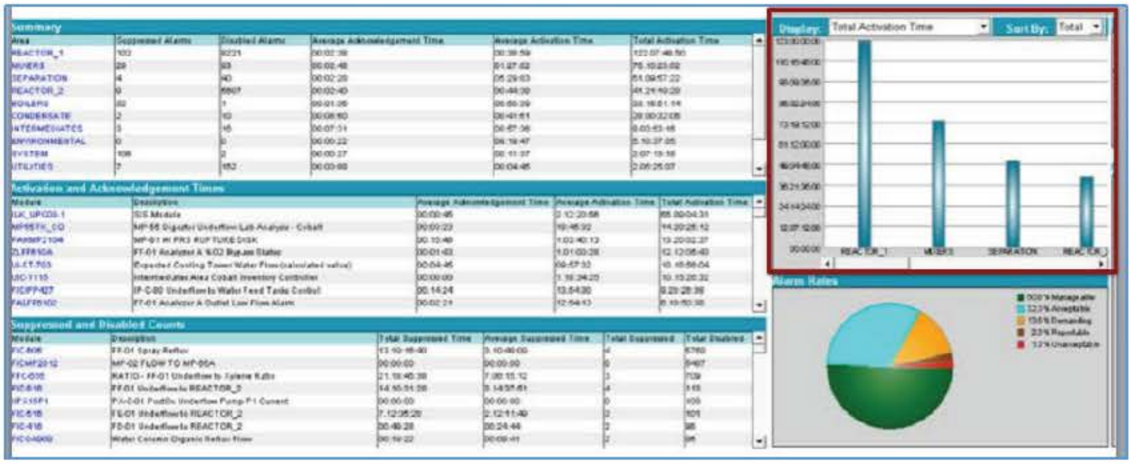
Classification. Purpose: Representation. Level of abstraction: Mechanism.

Context. A set of incoming alarms has been registered. The human operator would like to compare the number of alarms across different dimensions such as priority, typology or time.

Problem. *The human operator needs to compare the number of alarm activations across alarm dimensions.*

Solution. Use bar charts to represent absolute magnitudes of alarm activations. A bar chart is graphic figure in which numeric quantities are represented by the linear extent of parallel lines or bars, either horizontally or vertically. They can theoretically consist only a single data item, but in most cases are used to additionally compare the quantitative value of several entities with each other. All data items are measure on the same scale.

Known uses.



Annex Fig. A-33 An implementation of the Bar charts pattern: DeltaV system alarm statistics page (see a bar chart for alarm activation rates per device framed in red) [43]

Annex Fig. A-34 An implementation of the Bar charts pattern: IrisView system alarm dashboard which gives the operator a view of any alarms associated with the elements they are responsible for (see a bar chart for alarm activation rates per device framed in red) [65]

Rationale. Bar charts are, besides pie charts the most common data visualization technique and find wide usage in popular statistics [20]. They are useful to compare several quantitative entitles of a common class.

Relations.

- Specialization relationship: *RF(4) Trend displays*
- Generalization relationship: *RM(12) Histograms*

RM(12) Histograms

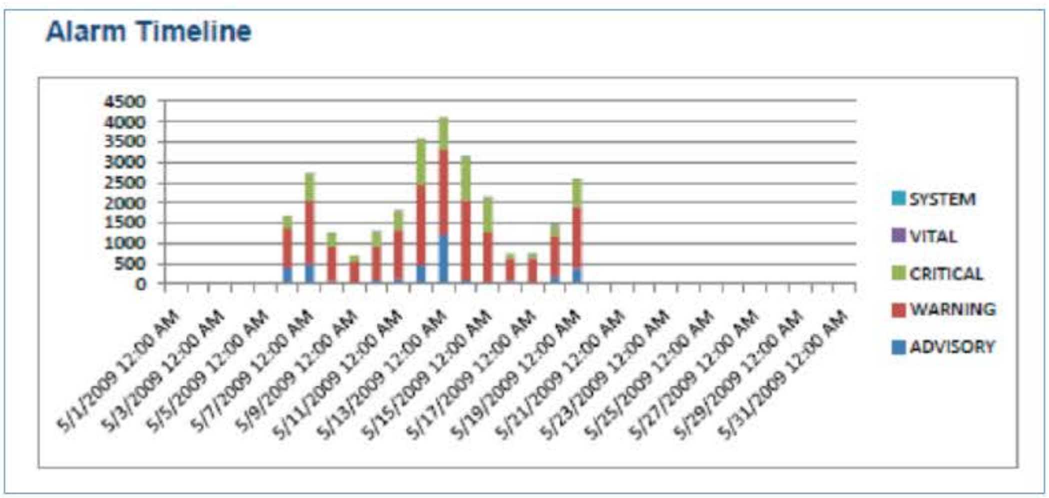
Classification. Purpose: Representation. Level of abstraction: Mechanism.

Context. A set of incoming alarms has been registered. The human operator would like to compare the number of alarm activations across different dimensions such as priority, typology or time. For each alarm dimension, he would need to know their different ranges.

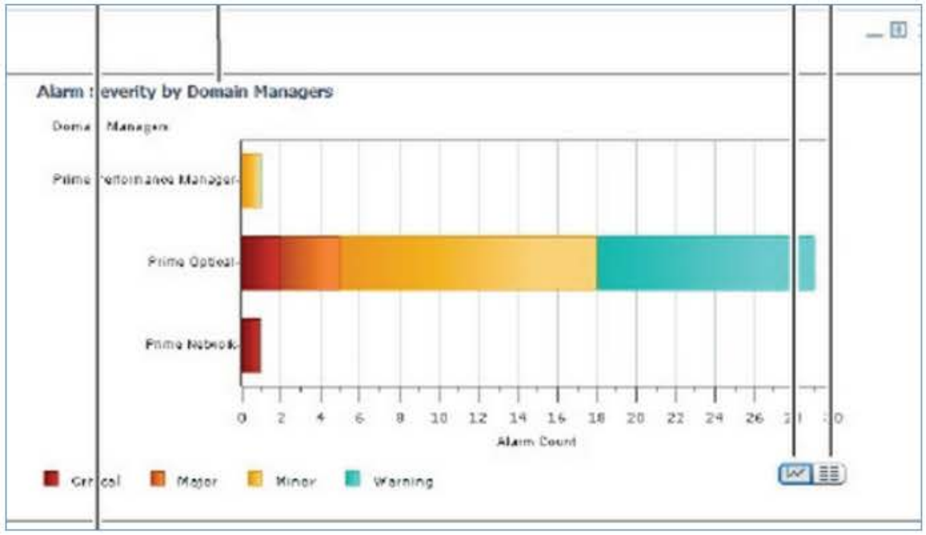
Problem. *The human operator needs to compare the number of alarm activations across alarm dimensions and considering data ranges.*

Solution. Use histograms to depict the frequency distribution of alarm activations. A histogram is a particular type of bar chart that groups data into ranges. Consequently, a rectangle is drawn with height proportional to the count and width equal to the bin size, so that rectangles abut each other.

Known uses.



Annex Fig. A-35 Implementation of the Histograms pattern: DeltaV system alarm summary report [43]



Annex Fig. A-36 Implementation of the Histograms pattern: Histogram provided by Cisco Prime Central system [28]

Rationale. Histograms give a rough sense of the density of the data, and often for density estimation [141].

Relations.

- Specialization relationship: *RM(11) Bar charts*

RM(13) PIE CHARTS

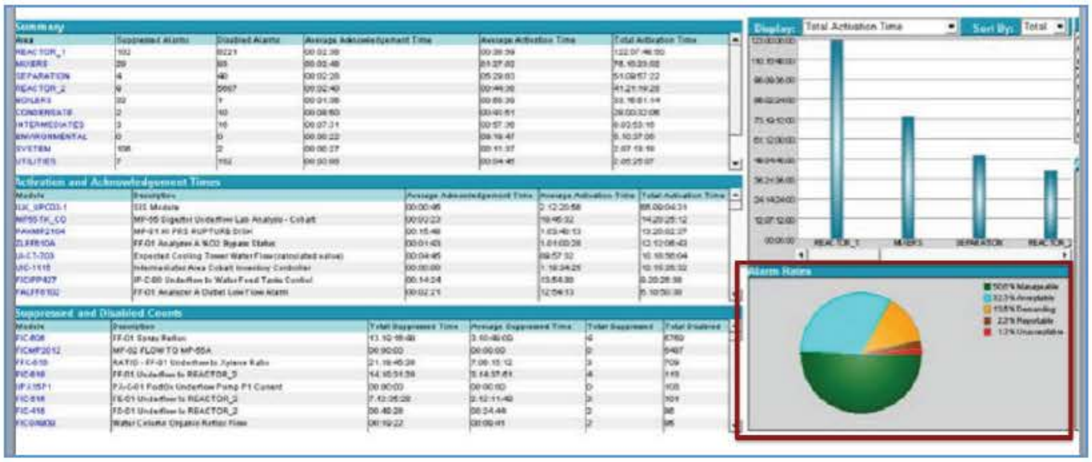
Classification. Purpose: Representation. Level of abstraction: Mechanism.

Context. A set of incoming alarms has been registered. The human operator would like to compare the distribution of alarms across different alarm dimensions such as priority or typology.

Problem. *The human operator needs to compare the distribution of alarm activations across different alarm dimensions in order to project future states of the controlled process.*

Solution. Use pie charts to show the alarm distribution. A pie chart is a circular object divided into multiple polar segments. It displays the relative magnitude of several quantitative values compared to each other. The full circle represents the total magnitude of this dataset, equal to 100 per cent, while each segment stands for the magnitude of one particular variable. Segment area, are length and arc angle of each segment are proportional to the value the segment represents.

Known uses.



Annex Fig. A-37 An implementation of the Pie charts pattern: DeltaV system alarm statistics page (see a pie chart for alarm activation rates per device framed in red) [43]

Annex Fig. A-38 An implementation of the Pie charts pattern: IrisView system alarm dashboard which gives the operator a view of any alarms associated with the elements they are responsible for (see a pie chart for alarm distribution by severity framed in red) [65]

Rationale. Pie charts give the reader a quick idea of the proportional distribution of data. The association between data and representation is evident: the bigger the piece, the larger the data chunk compared to the other ones.

Relations.

- Specialization relationship: *RF(4) Trend displays*

RM(14) LINEAR CHARTS

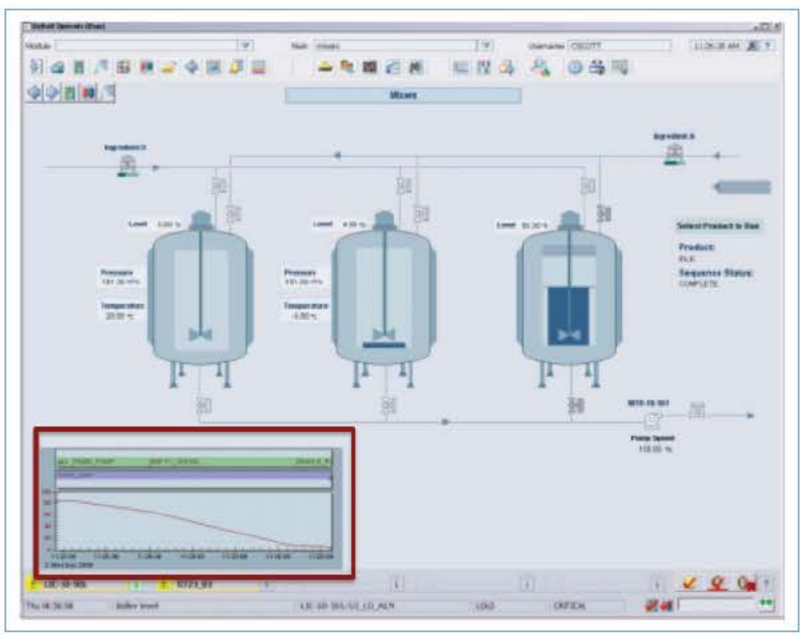
Classification. Purpose: Representation. Level of abstraction: Mechanism.

Context. A set of incoming alarms has been registered. The human operator would like to know the temporal evolution of alarms.

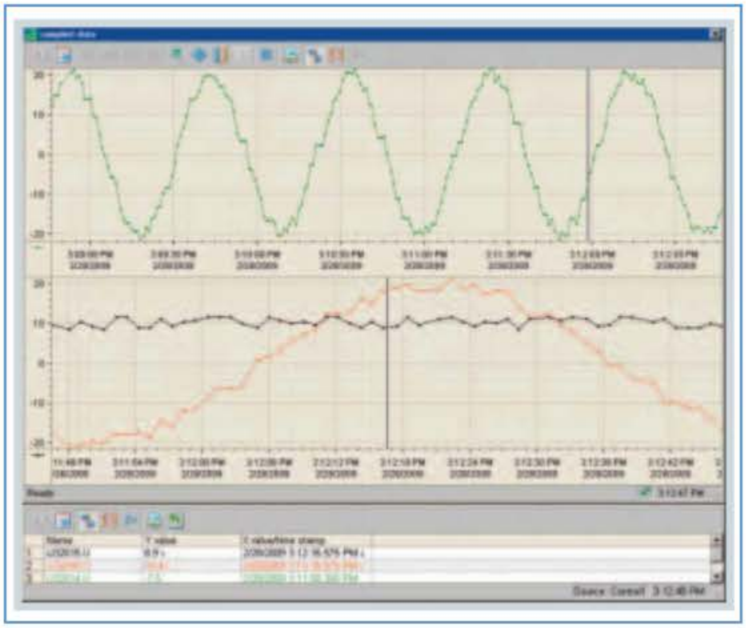
Problem. *The human operator needs to visualize how the alarm activations change over time.*

Solution. Use line charts to display the quantitative value of alarm activations over a continuous interval. In most cases, this interval is a time span, and the line chart describes how the object's variable changes over this time interval. Besides the individual values themselves the most significant information that can be derived from it is the gradient of the curve, which provides information about the intensity of the attribute's change over time. Also, minimum and maximum values can be easily identified from such a representation.

Known uses.



Annex Fig. A-39 An implementation of the Linear charts pattern: Embedded linear chart in the DeltaV process control system [43]



Annex Fig. A-40 An implementation of the Linear charts pattern: Linear chart provided by the SIMATIC PCS 7 Process Control system [121]

Rationale. The expressive power of this type of chart lies in the user's ability to interpret the development of a magnitude over an interval, such as time, by merely looking at the way the graph line moves from left to right.

Relations.

- Specialization relationship: *RF(4) Trend displays*

INTERACTION PATTERNS

IF(1) Direct manipulation

Classification. Purpose: Interaction. Level of abstraction: Feature.

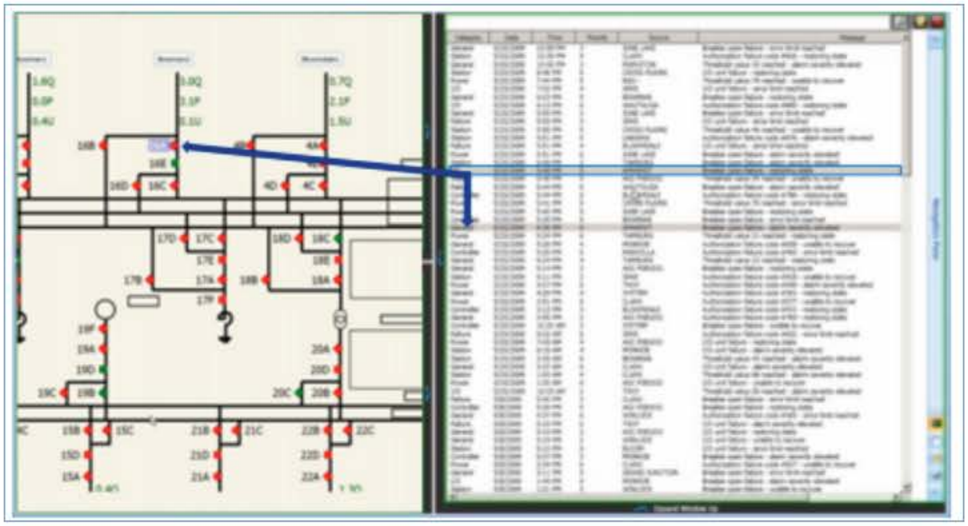
Context. Not all controlled process conditions require the same degree of response or attention by a human operator. Accordingly, the human operator would like accessing, manipulating and navigating alarm displays in order to access to the most relevant alarm information for the current status of the controlled process.

Problem. *The human operator needs to access, manipulate and navigate alarm displays to easily access to the most relevant alarm information for the current status of the controlled process.*

Solution. Use direct manipulation to allow operators to provide inputs to an alarm display, receive information from it, and manage the tasks associated with access and control of alarm information. Using a pointing device to manipulate the graphical object, causing the computer operations to be performed on the object or information it represents, usually provides input. Feedback is represented by a change in the graphic object.

Known uses.

Annex Fig. A-41 An implementation of the Direct manipulation pattern: Use of sliders (framed in red) and buttons (pointed by the arrow) to allow the human operator to provide inputs to the ELSA control system interface [42]



Annex Fig. A-42 An implementation of the Direct manipulation pattern: Selection of one alarm in the alarm list by using a pointing device (right side) causing the alarm is highlighted in the diagram (left side) [1]

Rationale. Having real-world metaphors for objects and actions can make it easier for a user to learn and use an interface (some might say that the interface is more natural or intuitive), and rapid, incremental feedback allows a user to make fewer errors and complete tasks in less time, because they can see the results of an action before completing the action, thus evaluating the output and compensating for mistakes [20].

- **Relations.**
  - Generalization relationship: *IM(3) Dynamic queries, IM(4) Brushing and linking*

IF(2) Display resolution management

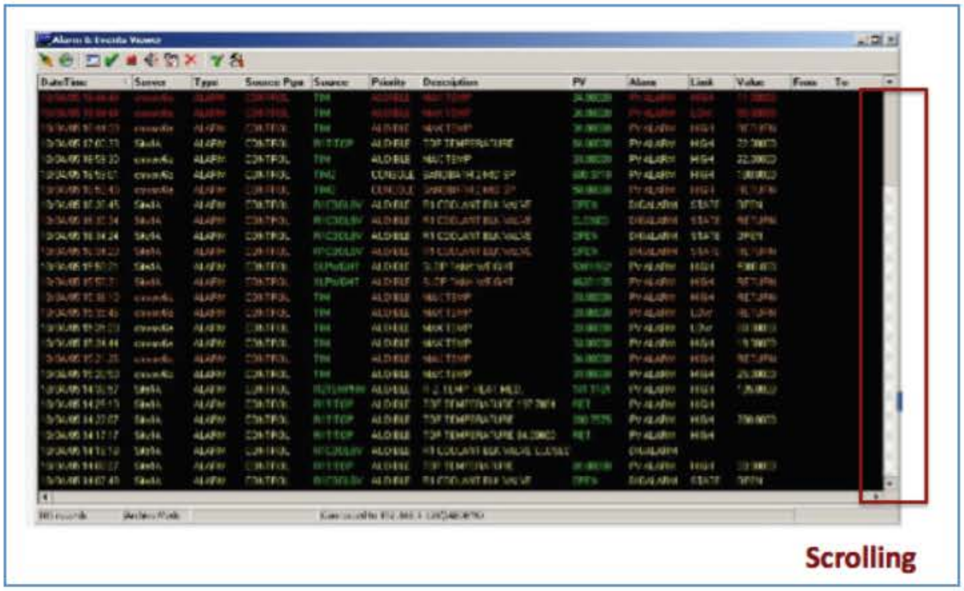
Classification. Purpose: Presentation. Level of abstraction: Feature.

Context. Display pages are sometimes too large to be viewed all at once from a single alarm display screen with a level of resolution adequate for operator's tasks.

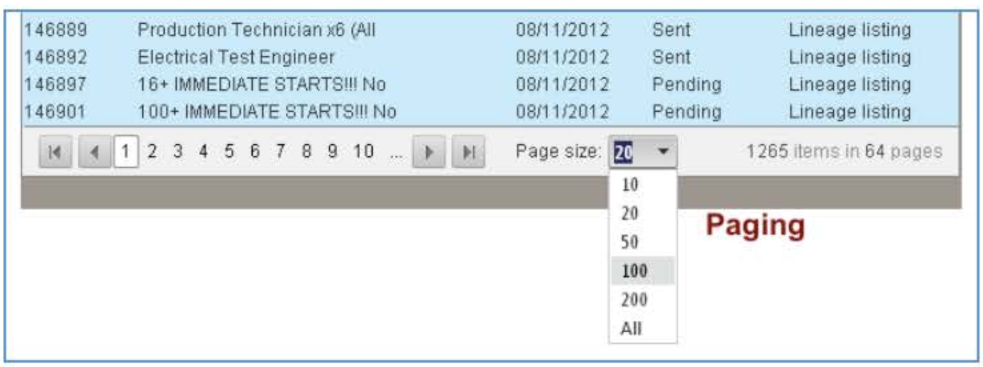
Problem. *The human operator needs to be able to navigate through large alarm displays.*

Solution. Use mechanisms that allow the operator specify a different degree of interest in different parts of the information. These mechanisms should allow alarm information to be moved or spatially partitioned. They should also be designed to facilitate target detection. For example, these mechanisms should be sufficiently slow when approaching the target so the operator can recognize the target.

Known uses.



Annex Fig. A-43 An implementation of the Display resolution management pattern: Use of scrolling applied to an alarm list [107]



Annex Fig. A-44 An implementation of the Display resolution management pattern: Use of numeric paging applied to an alarm list [1]

Rationale. In contrast to filtering techniques, these mechanisms combine the visualization of detail alarm information with a representation of an overview of contextual data or even the entire dataset. Discriminating interesting from contextual information is agreed as a suitable and effective approach to achieve visual scalability in visualization [20].

Relations.

- Generalization relationship*: IM(5) Zooming, IM(6) Panning, IM(7) Scrolling, IM(8) Paging,* and *IM (9) Distortion.*

IM(3) DYNAMIC QUERIES

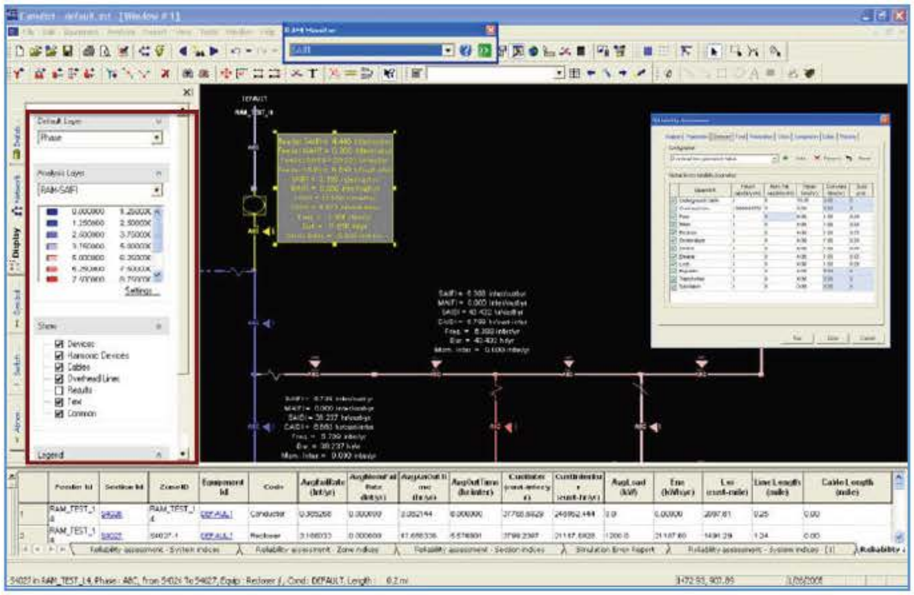
Classification. Purpose: Interaction. Level of abstraction: Mechanism.

Context. Not all controlled process conditions require the same degree of response or attention by a human operator. Accordingly, the human operator would like accessing, manipulating and navigating alarm displays in order to access to the most relevant alarm information for the current status of the controlled process.

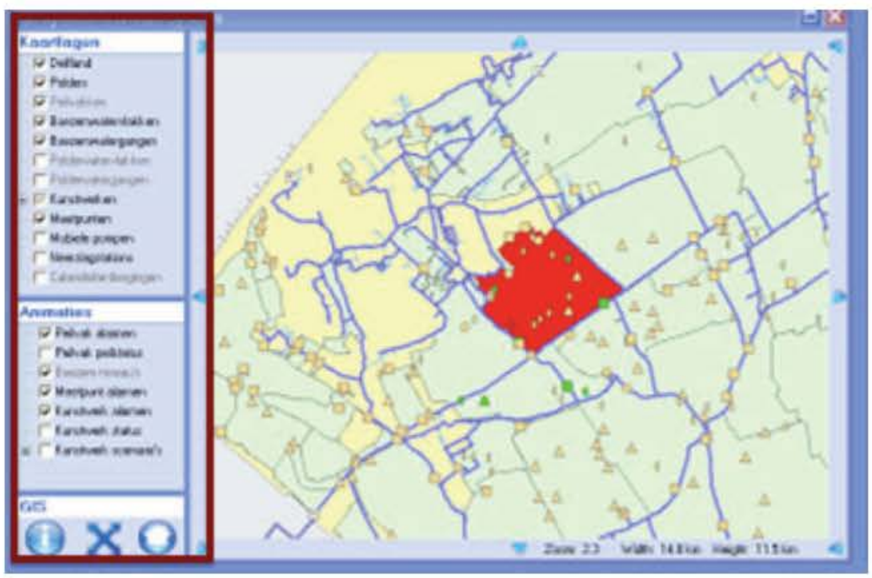
Problem*. The human operator needs to access only a certain part of the alarm information.*

Solution. Represent graphically the request of the human operator by using dynamic queries. Dynamic queries provide a graphical visualization of a database and searching results. Using a pointing device usually provides input by the human operator. They allow to clearly dividing the alarm information into distinct separated categories or layers, such as alarms placed on an integrated display. They work instantly within a few milliseconds as users adjusts sliders or select buttons to form simple queries. In this way, they let the user switch filters on and off by preference: if the operator wants to explore only a certain part of the alarm information belonging to one specific category, he can deselect all the other filters to clean up the display.

Known uses.



Annex Fig. A-45 An implementation of the Dynamic queries pattern: The CYMDIST Distribution Analysis software complements the geographic view of an electrical network with a set of filters (framed in red) [30]



Annex Fig. A-46 An implementation of the Dynamic queries pattern: GIS viewer applies dynamic queries to a map surface [53]

Rationale. A dynamic query function provides the user with instant gratification for his input effort. The user gets immediate feedback to his query from the very first keystroke, and quickly notices if he's on the wrong track. The possible result span is not limited gradually, but this technique also helps to point out spelling mistakes and impossible criteria combinations from the outset.

Relations.

- Specialization relationship. *IF(1) Direct manipulation*
- Combination relationship: *RF(3) Integrated displays*

IM(4) Brushing and linking

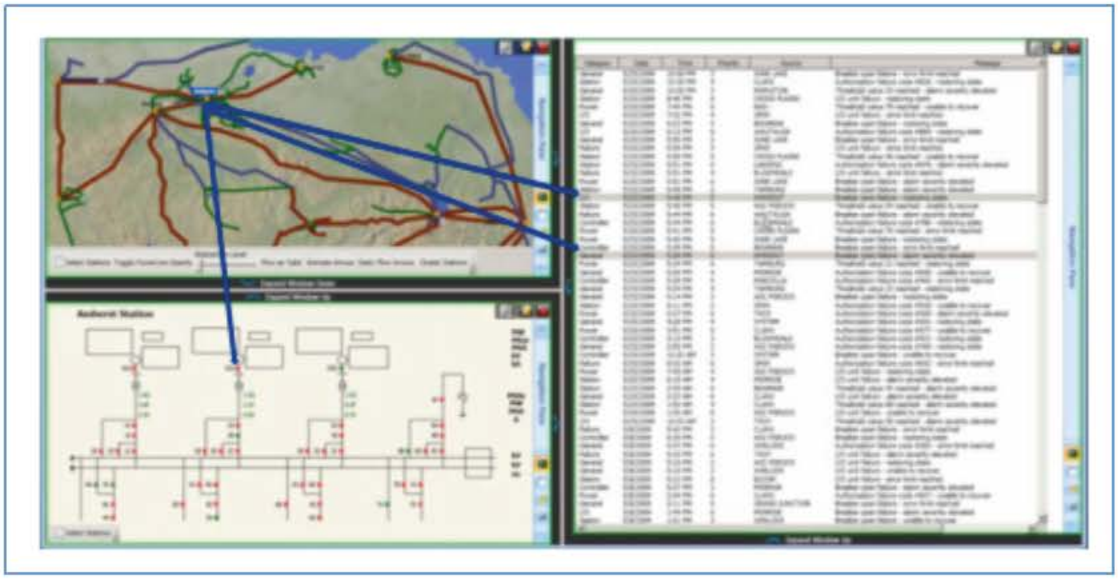
Classification. Purpose: Interaction. Level of abstraction: Mechanism.

Context. Not all controlled process conditions require the same degree of response or attention by a human operator. Accordingly, the human operator would like accessing, manipulating and navigating alarm displays in order to access to the most relevant alarm information for the current status of the controlled process.

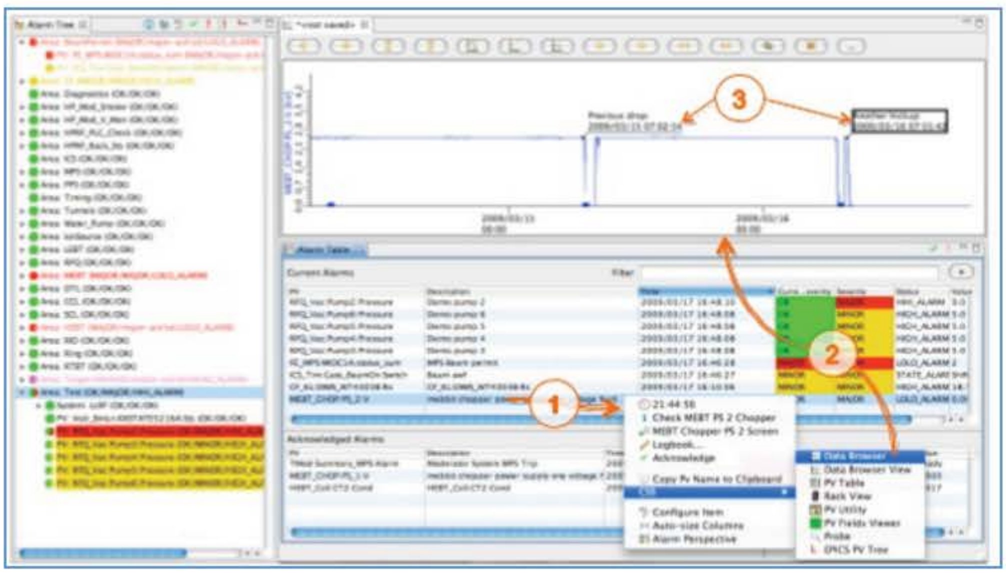
Problem. *The human operator needs to easily access to the most relevant alarm information for the current status of the controlled process.*

Solution. Create a tightly coordination between alarm displays at different levels of detail but also between displays at the same level of detail by applying brushing and linking. Brushing and linking refers to the connection of two or more views of the same information, such that a change to the representation in one view affects the representation in the other. Specifically, brushing refers to highlighting, for example selected data, in one view, in other connected data representations. Linking refers to a change of parameters in one data representation being reflected in other connected data representations.

Known uses.



Annex Fig. A-47 An implementation of the Brushing and linking pattern: Substation selection in geographical overview of a power grid infrastructure highlights alarms in alarm list and displays the corresponding detailed views with the alarming devices highlighted [87]



Annex Fig. A-48 An implementation of the Brushing and linking pattern: Multiple coordinated views by using brushing and linking [87]

Rationale. Brushing and linking has been characterized as an effective technique for performing visual exploration and analysis of large, structured data sets [141]. It allows identifying patterns and outliers across displays.

Relations.

- Specialization relationship. *IF(1) Direct manipulation*
- Combination relationship*: PM(5) Overview and detail*

IM(5) Zooming

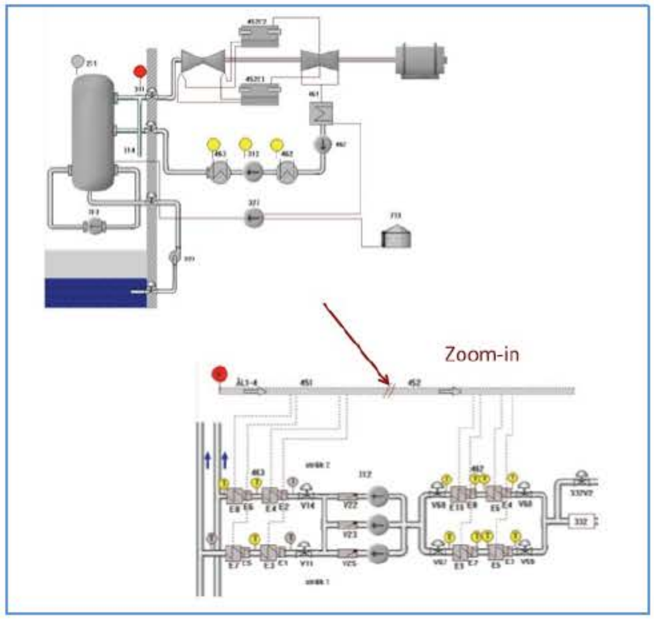
Classification. Purpose: Presentation. Level of abstraction: Mechanism.

Context. Display pages are sometimes too large to be viewed all at once from a single alarm display screen with a level of resolution adequate for operator's tasks.

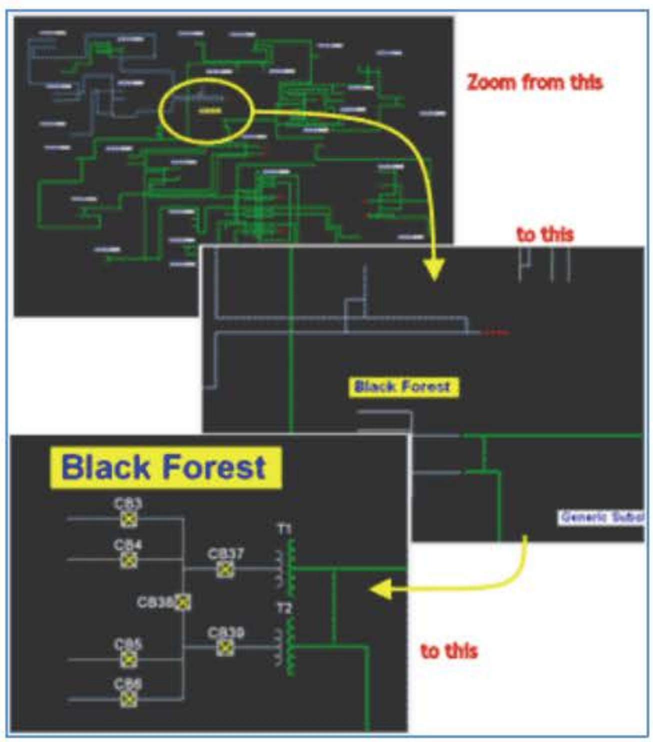
Problem*. The human operator needs to move rapidly and fluidly between levels of detail of alarm information.*

Solution. Support focused and contextual views based on zooming, which involves a temporal separation between these views. Zooming is based on a camera analogy; the action is analogous to changing the focal length of a camera lens. It is possible to magnify a decreasing fraction (or vice versa) of an element under the constraint of a viewing frame of constant size. Zoom-in is similar to moving closer to an object while zoom-out is similar to moving further away from it. Because the size of the display screen is fixed, the effect of zooming-in is to show a smaller area of the display page at a higher magnification; the effect of zooming-out is to show a larger area at lower magnification.

Known uses.



Annex Fig. A-49 An implementation of the Zooming pattern: Applying zooming to a power plant map [5]



Annex Fig. A-50 An implementation of the Zooming pattern: Applying zooming techniques to the Catapult iPower SCADA operational environment for electric control room SCADA [23]

Rationale. Zooming facilitates two different cognitive tasks [124]. With zooming-in, extraneous information is removed from the visual field, perhaps resulting in a more manageable view, whereas zooming-out reveals hidden information, often context that is already known but perhaps cannot be recalled. It often allows a user to rediscover their location in an information space and to integrate a new context within a mental model.

Relations

- Specialization relationship*: IF(2) Display resolution management*
- Alternative relationship*: IM(9) Distortion*
- Combination relationship: *RF(3) Integrated displays*

IM(6) Panning

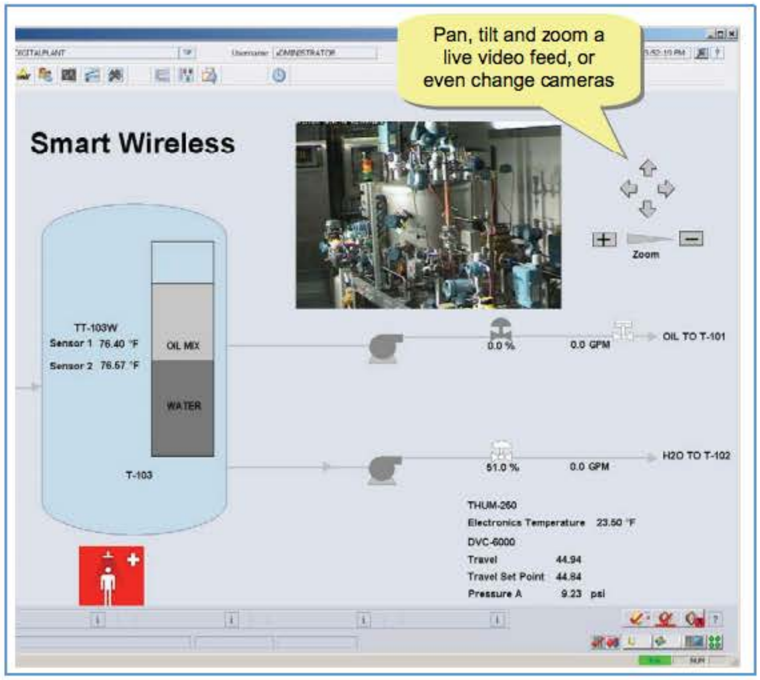
Classification. Purpose: Presentation. Level of abstraction: Mechanism.

Context. Display pages are sometimes too large to be viewed all at once from a single alarm display screen with a level of resolution adequate for operator's tasks.

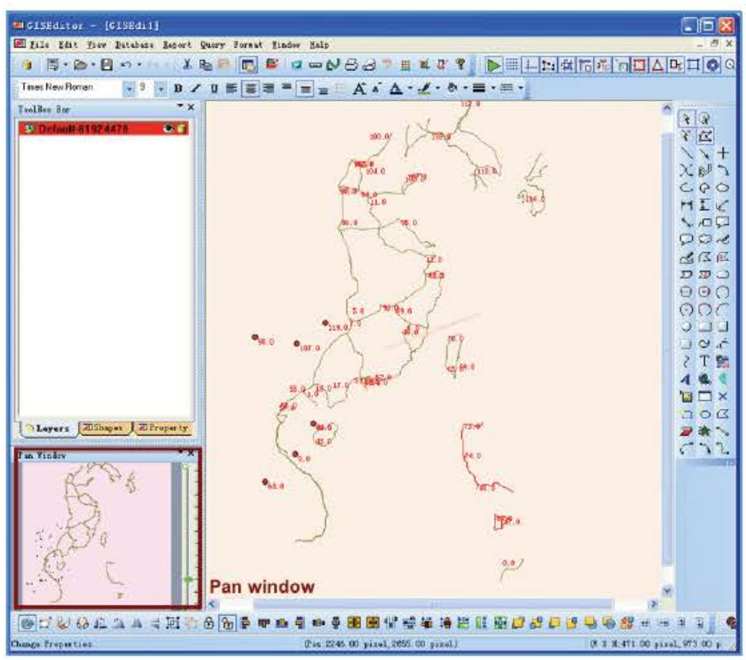
Problem*. The human operator needs to navigate across alarm activations that do not fit the display as a whole.*

Solution. Change the section of the area to be displayed in the alarm display based on panning. Panning allows moving a viewing frame over a display space of greater size. However, there are several ways to let the operator pan the data space. The classic variation of it uses a set of arrow buttons that move the data space by a certain value to the left, right, top or bottom. Another variation has a small inset overview region, which includes an interactive rectangular sub-region, called pan window, which corresponds to the area shown in the detailed view. Often used as a redundant feature along with buttons is panning by drag and drop. When the mouse pointer moves into viewport, mark it as a dragging tool by changing its appearance from an arrow to an open hand. The distinction from scrolling is one of perspective; panning is the opposite of scrolling. When panning, the viewer perceives the displayed material as being stationary while the viewing area of the display screen moves across it. In applications where a user moves a cursor freely about a page of displayed data, panning should be adopted rather than scrolling as the conceptual basis of display framing.

Known uses.



Annex Fig. A-51 An implementation of the Panning pattern: Use of a set of arrows buttons in the DeltaV process control system interface [43]



Annex Fig. A-52 An implementation of the Panning pattern: Use of a pan window (framed in red) at the bottom of the interface [39]

Rationale. Whether moving along incrementally by clicking arrow buttons, or dragging the content freely around, it forces to provide the user a way to move the content pane as it will not fit into the display space at once [124].

Relations.

- Specialization relationship*: IF(2) Display resolution management*
- Combination relationship: *RF(3) Integrated displays*

IM(7) Scrolling

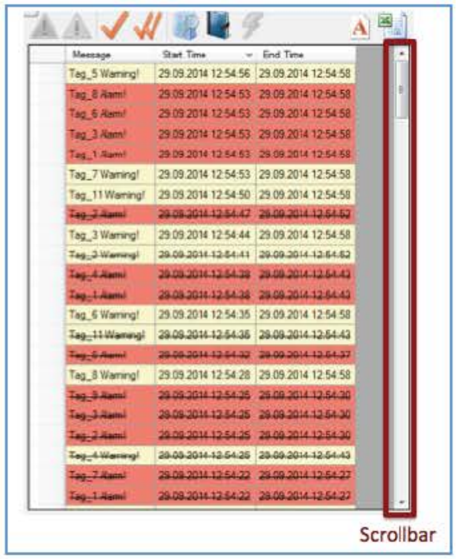
Classification. Purpose: Presentation. Level of abstraction: Mechanism.

Context. Display pages are sometimes too large to be viewed all at once from a single alarm display screen with a level of resolution adequate for operator's tasks.

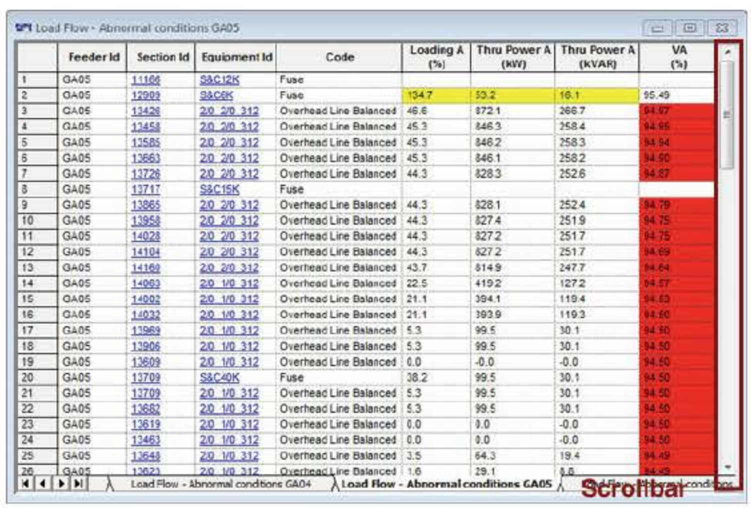
Problem*. The human operator needs to be able to navigate through a list of alarms.*

Solution. Slide elements vertically or horizontally based on scrolling. Scrolling is a display framing technique that allows the user to view a display as moving behind a fixed frame. A scroll bar, keyboard arrow keys might perform scrolling, and keystroke commands. The scrolling action is typically combined with alarm lists and causes the data displayed at one end of the screen to move across it, toward the opposite end. When the data reach the opposite edge to the screen they are removed. Thus, old data are removed from one end while new data are added at the other. This creates the impression of the display space being on an unwinding scroll, with only a limited portion being visible at any time from the screen. Displays may be scrolled in the top-bottom direction, the left-right direction, or both.

Known uses.



Annex Fig. A-53 An implementation of the Scrolling pattern: Alarm list with scroll bar vertical enabled from a SCADA system interface [113]



Annex Fig. A-54 An implementation of the Scrolling pattern: Alarm list with scroll bar vertical enabled from a CYMDIST Distribution Analysis software [30]

Rationale. Scrolling is faster for users than clicking. Users can see all content in order on the display without needing to click any links.

Relations.

- Combination relationship: *RF(5) Lists*
- Specialization relationship*: IF(2) Display resolution management*

- Alternative relationship: *IM(8) Paging*

IM(8) Paging

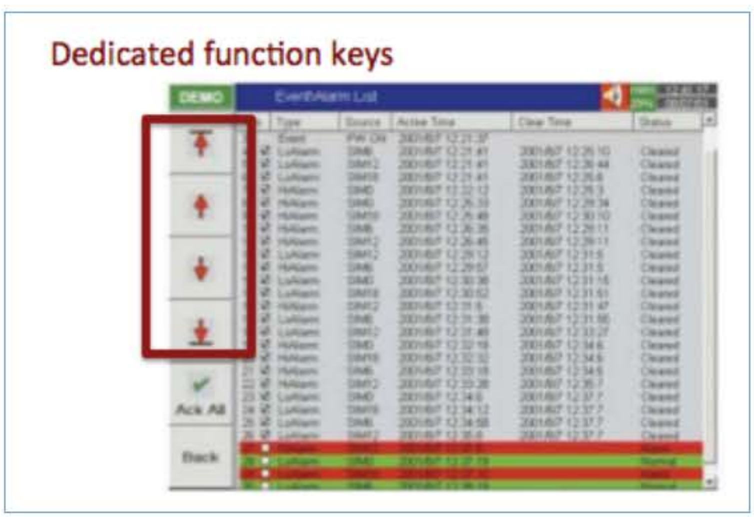
Classification. Purpose: Presentation. Level of abstraction: Mechanism.

Context. The available display space is often smaller than the area populated with alarm activations. The human operator would like to explore an area with alarm activations that is larger than the alarm display.

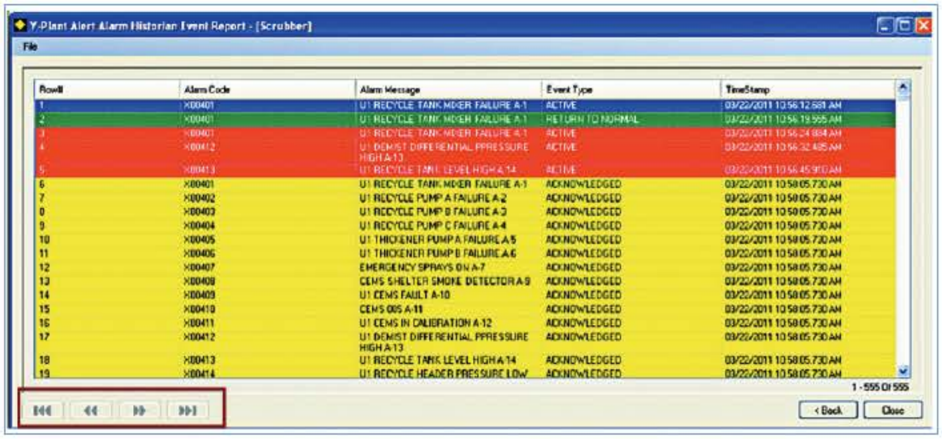
Problem. *The human operator needs to be able to navigate through a list of alarms.*

Solution. Divide alarm activations into a set of display-size pages based on paging. Paging is a display framing technique that allows the user to view a display as a set of display-size pages that are accessed in discrete steps. Thus, rather than being presented as a scroll, the display page is presented as a set of discrete pages. These pages are often accessed sequentially. Paging should be available by means of moving a page icon on the scroll bar, or by the use of a dedicated function key for paging forward and a dedicated function key for paging back through a file. Scrolling makes it difficult to read alarm messages, especially when many alarms are coming in. Paging is preferred to view alarm lists in such conditions. However, it should not be used when searching through continuous text data.

Known uses.



Annex Fig. A-55 An implementation of the Paging pattern: Use of dedicated function keys in a vertical bar for paging from a CAS data logger [22]



Annex Fig. A-56 An implementation of the Paging pattern: Use of dedicated function keys in a horizontal bar for paging provided by the Y-Plant Alert System [146]

Rationale. Paging give users an easy mean to move back and forth over displayed material when requested data exceeds the capacity of a single display frame [37].

Relations.

- Combination relationship: *RF(5) Lists*
- Specialization relationship*: IF(2) Display resolution management*
- Alternative relationship*: IM(7) Scrolling*

IM(9) Distortion

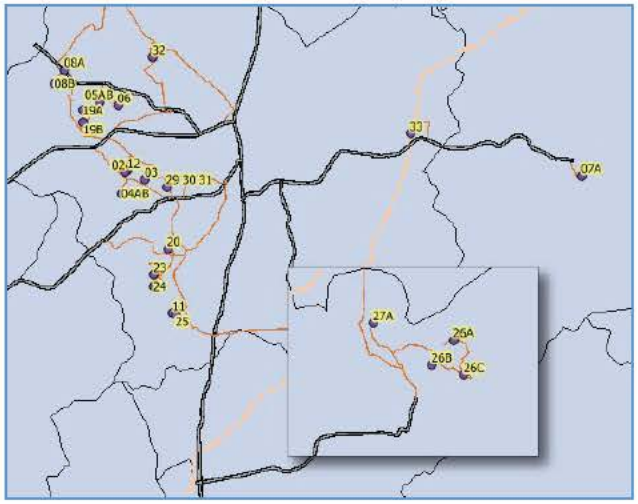
Classification. Purpose: Presentation. Level of abstraction: Mechanism.

Context. The available display space is often smaller than the area populated with alarm activations. The human operator would like to explore an area with alarm activations that is larger than the alarm display.

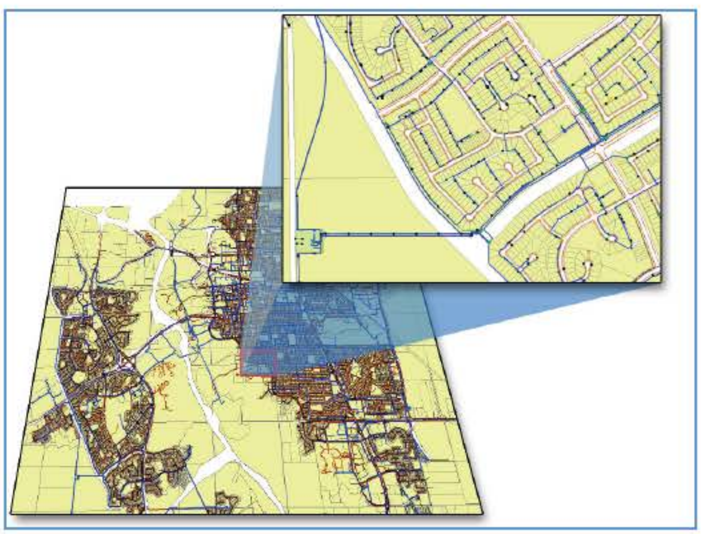
Problem*. The human operator needs to move rapidly and fluidly between levels of detail of alarm information.*

Solution. Support focused and contextual views based on distortion, which integrates focus and context views into a single display where all parts are concurrently visible: the focus is displayed seamlessly within its surrounding context. Distortion presents the focus area at a higher magnification than the rest of the display. The result is a distorted view of the large display page because different parts of it give the user contextual information. Key features of the unmagnified global structure inform the user of the existence and location of other parts of the information structure and support the interpretation of local details.

Known uses.



Annex Fig. A-57 An implementation of the Distortion pattern: Geographical map with a magnifying glass effect around a specific area [148]



Annex Fig. A-58 An implementation of the Distortion pattern: Geographical map of power generation with a magnifying effect around a specific area [123]

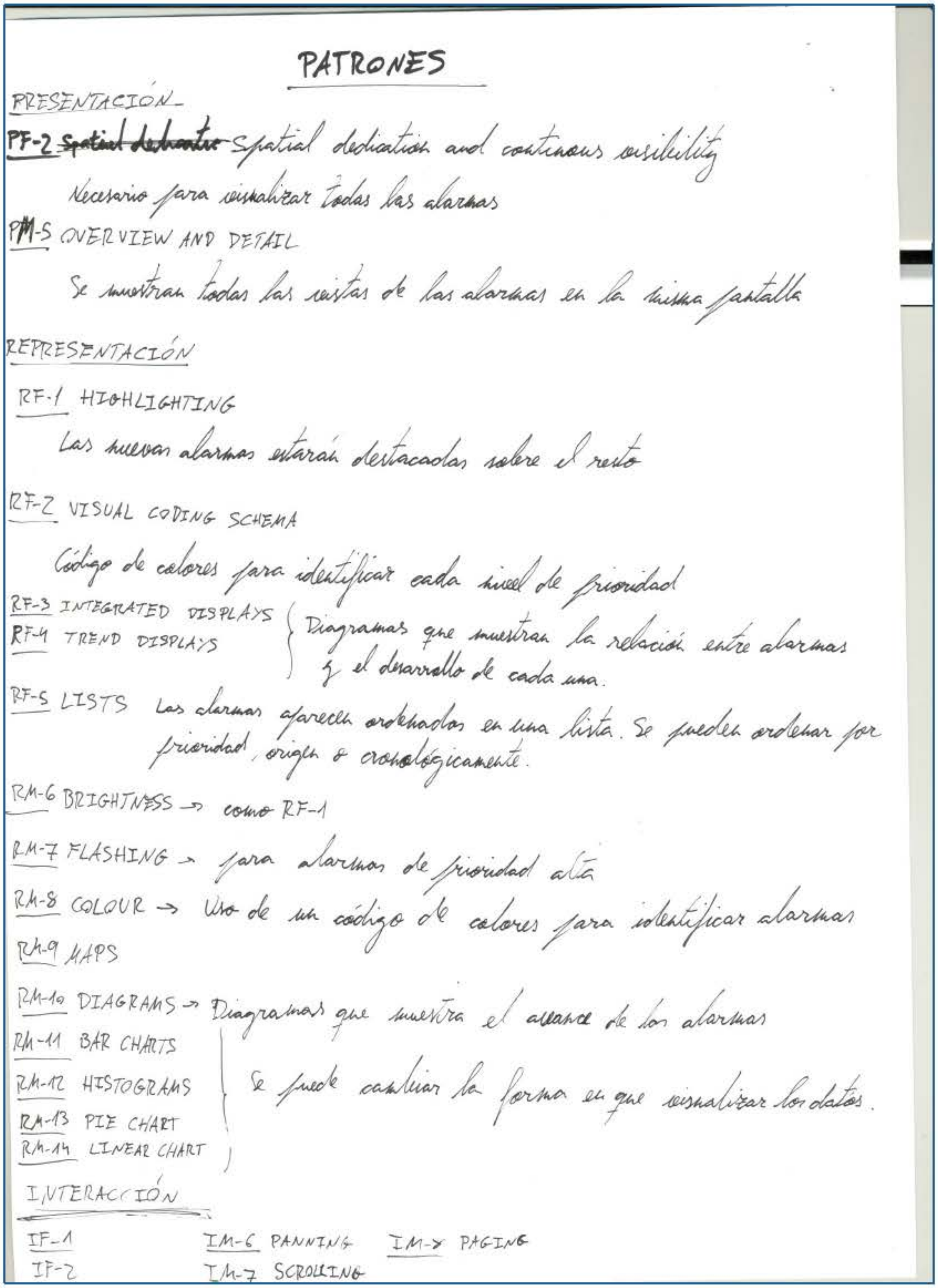
Rationale. Distortion-oriented techniques prioritize the use of display space based on importance, while providing context to the rest of the display [37]. They decrease the short-term memory load associated with assimilating distinct views of a system, and thus potentially improve user ability to comprehend and manipulate the information.

Relations.

- Combination relationship: *RF(3) Integrated displays*
- Specialization relationship*: IF(2) Display resolution management*
- Alternative relationship*: IM(5) Zooming*

Annex B - Collection of sketches (First round of the Experimental Evaluation)

Annex Fig. B-1 Sketch generated by participant 1 using the design pattern language



PATRONES

PRESENTACIÓN

PF-2 ~~Spatial dedicatio~~ Spatial dedication and continous visibility

Necesario para visualizar todas las alarmas

PM-5 OVERVIEW AND DETAIL

Se muestran todas las vistas de las alarmas en la misma pantalla

REPRESENTACIÓN

RF-1 HIGHLIGHTING

Las nuevas alarmas estarán destacadas sobre el resto

RF-2 VISUAL CODING SCHEMA

Código de colores para identificar cada nivel de prioridad

RF-3 INTEGRATED DISPLAYS
RF-4 TREND DISPLAYS
} Diagramas que muestran la relación entre alarmas y el desarrollo de cada una.

RF-5 LISTS Las alarmas aparecen ordenadas en una lista. Se pueden ordenar por prioridad, origen o cronológicamente.

RM-6 BRIGHTNESS → como RF-1

RM-7 FLASHING → para alarmas de prioridad alta

RM-8 COLOUR → Uso de un código de colores para identificar alarmas

RM-9 MAPS

RM-10 DIAGRAMS → Diagramas que muestra el avance de las alarmas

RM-11 BAR CHARTS
RM-12 HISTOGRAMS
RM-13 PIE CHART
RM-14 LINEAR CHART
} Se puede cambiar la forma en que visualizar los datos.

INTERACCIÓN

IF-1 IM-6 PANNING IM-8 PAGING

IF-2 IM-7 SCROLLING

Annex Fig. B-2 Description of the design patterns applied to create the sketch by participant 1

Annex Fig. B-3 Sketch generated by participant 2 using the design pattern language

.PATRONES.

PF (3) Primer nivel de detalle en la seccion 1 y 2 mientras que en la 3 se muestra toda la información cuando se selecciona una alarma que sería información secundaria. PF(4)

PM (5) La overview se muestra en la sección 2 y 5

PM(6) Se muestran metricas en las secciones 2 y 5 donde las alarmas se disponen en niveles de prioridad y tipos o canales

RF(1) Cuando se selecciona una alarma en el mapa o en la sección 1, 2 ó 5 se marcará en el mapa aumentando el tamaño, en la sección 1 con sombras y en las secciones 2 y 5 con una flecha inferior.

RF(5) Lists lista cronológica en sección 1.

RM(8) La prioridad se marcará con colores en todos los lugares donde aparecen las alarmas. El resto de colores serán blancos y grises para identificar mejor las alarmas.

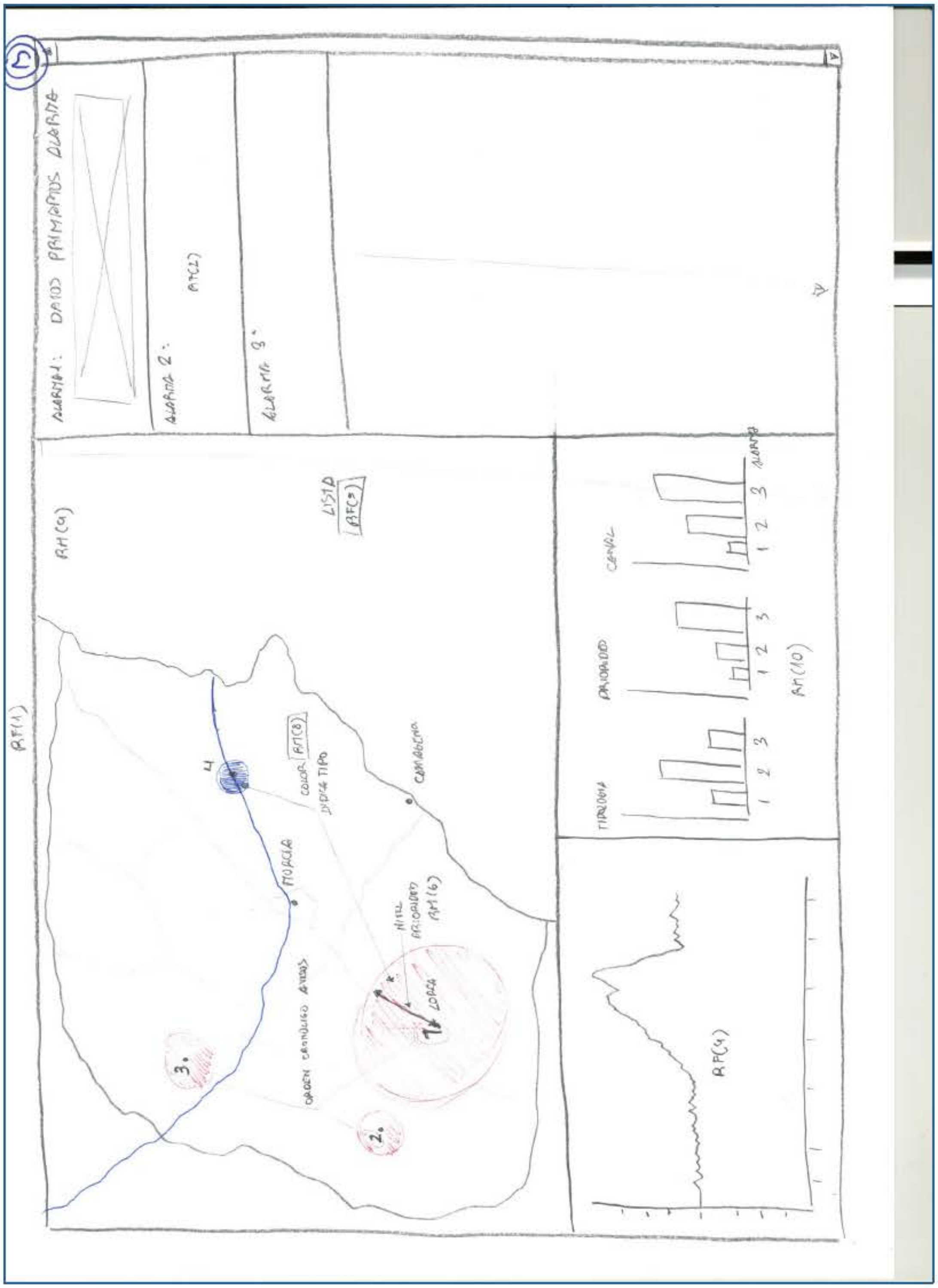
RM(9) El mapa es la sección principal de la interfaz proporciona una visión espacial de las alarmas

RM(11) Badges, se emplean para comparar el nº de alarmas de cada tipo y canal. En la sección 6.

IM(5) Dos opciones de zoom para el mapa

En las secciones 2 y 5 las alarmas se almacenan en tarjetas.

Annex Fig. B-4 Description of the design patterns applied to create the sketch by participant 2



Annex Fig. B-5 Sketch generated by participant 3 using the design pattern language

DISEÑO VISUALIZACIÓN DE ALARMA

PF(1): Aplicamos el patrón para la distribución de alarmas. La prioridad en este caso se encuentra en visualizar rápidamente en el mapa donde se sitúa la alarma por ello ocupa la mayor parte del espacio de la interfaz.

PF(2): Aplicamos el patrón en toda la interfaz. No existen ventanas emergentes ya que podrían distraer la atención de las alarmas. Es una interfaz plana. Está relacionado con el PM(6) ordenando por cronología el orden de alarmas.

PF(3). necesitamos mostrar el un detalle alto de la representación de las alarmas. Debido a esto utilizamos el patrón RF(5) para mostrar con una lista información específica de cada alarma (por ejemplo, nº avisos, coordenadas, prioridad de alarma, etc). También está relacionado con el RF(3), ya que la lista en la parte derecha muestra información entre las alarmas.

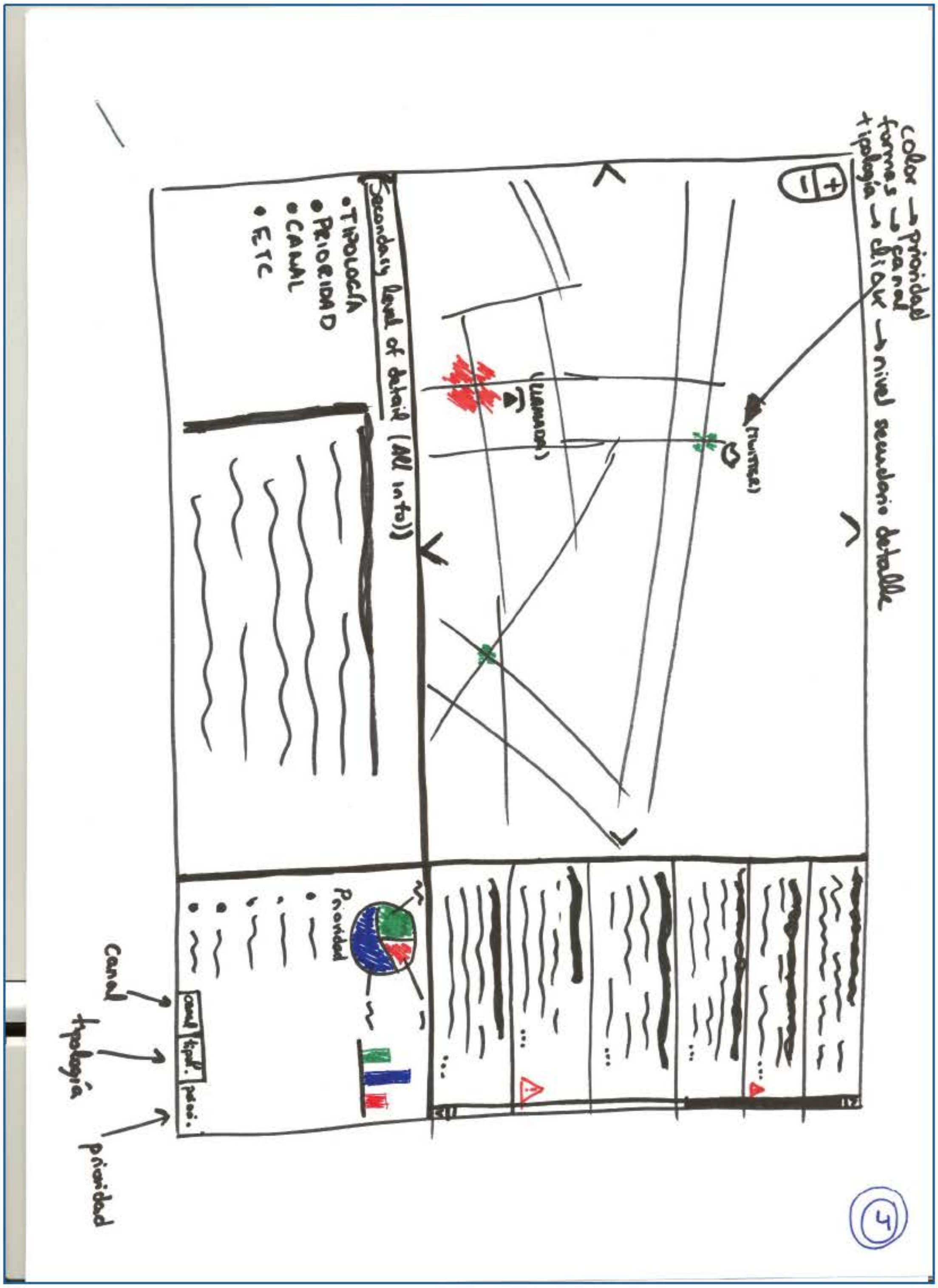
RF(4): mediante un gráfico disponemos los datos concretos de una alarma y ayuda a predecir situaciones futuras.

RM(10): mediante diagramas vemos para cada alarma, su tipología, su canal y su prioridad.

RM(6): hemos utilizado luz en el mapa para medir el nivel de prioridad entre alarmas.

RM(8): hemos utilizado colores para distinguir tipo de alarma en el mapa.

Annex Fig. B-6 Description of the design patterns applied to create the sketch by participant 3



Annex Fig. B-7 Sketch generated by participant 4 using the design pattern language

PATRONES

PRESENTACIÓN

- PF(2) Details hierarchy. Para tener una estructura de más general a más específico. → y al timeline
- PF(3) Primary level of detail. Aplicado en la zona del mapa, donde la info. es más general.
- PF(4). Secondary level of detail. Aplicado a la zona de debajo del mapa, donde se muestra toda la info.
- PM(5) Overview and detail. Para la estructura descrita anteriormente.

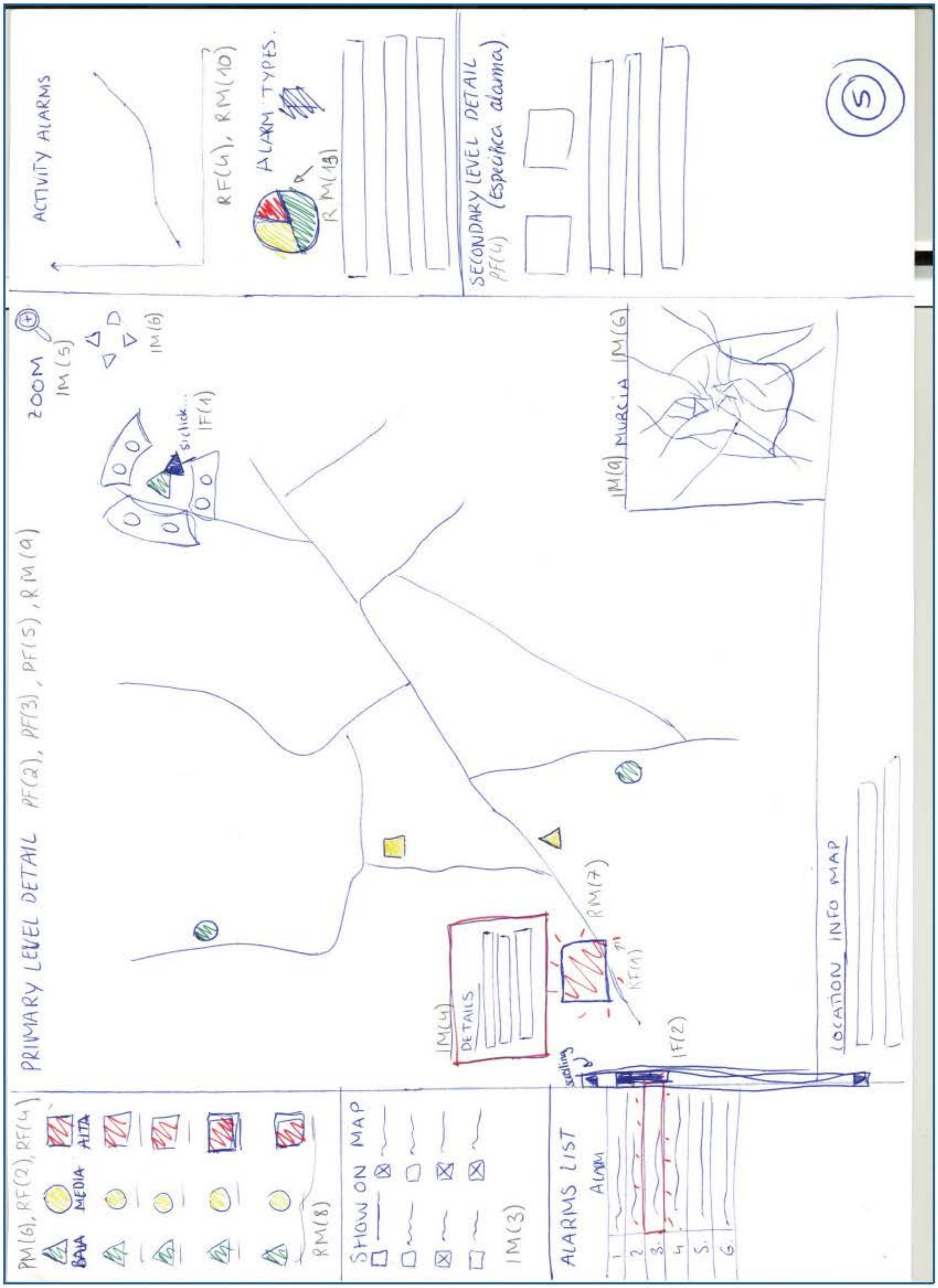
REPRESENTACIÓN

- RF(1) Highlighting. Para distinguir las alarmas con más prioridad de una forma "instintiva".
- RF(2) Visual coding schema. Para distinguir la prioridad con colores, el canal por los símbolos.
- RF(5) Lists. Aplicada una lista ordenada cronológicamente (timeline) para poder visualizar las alarmas en el tiempo en que van apareciendo.
- RM(8) Colour. Para diferenciar prioridades.
- RM(9) Maps. Para poder visualizar las alarmas en un contexto físico-espacial.
- RM(12) Histograms, RM(13) Pie charts. Para visualizar los datos.

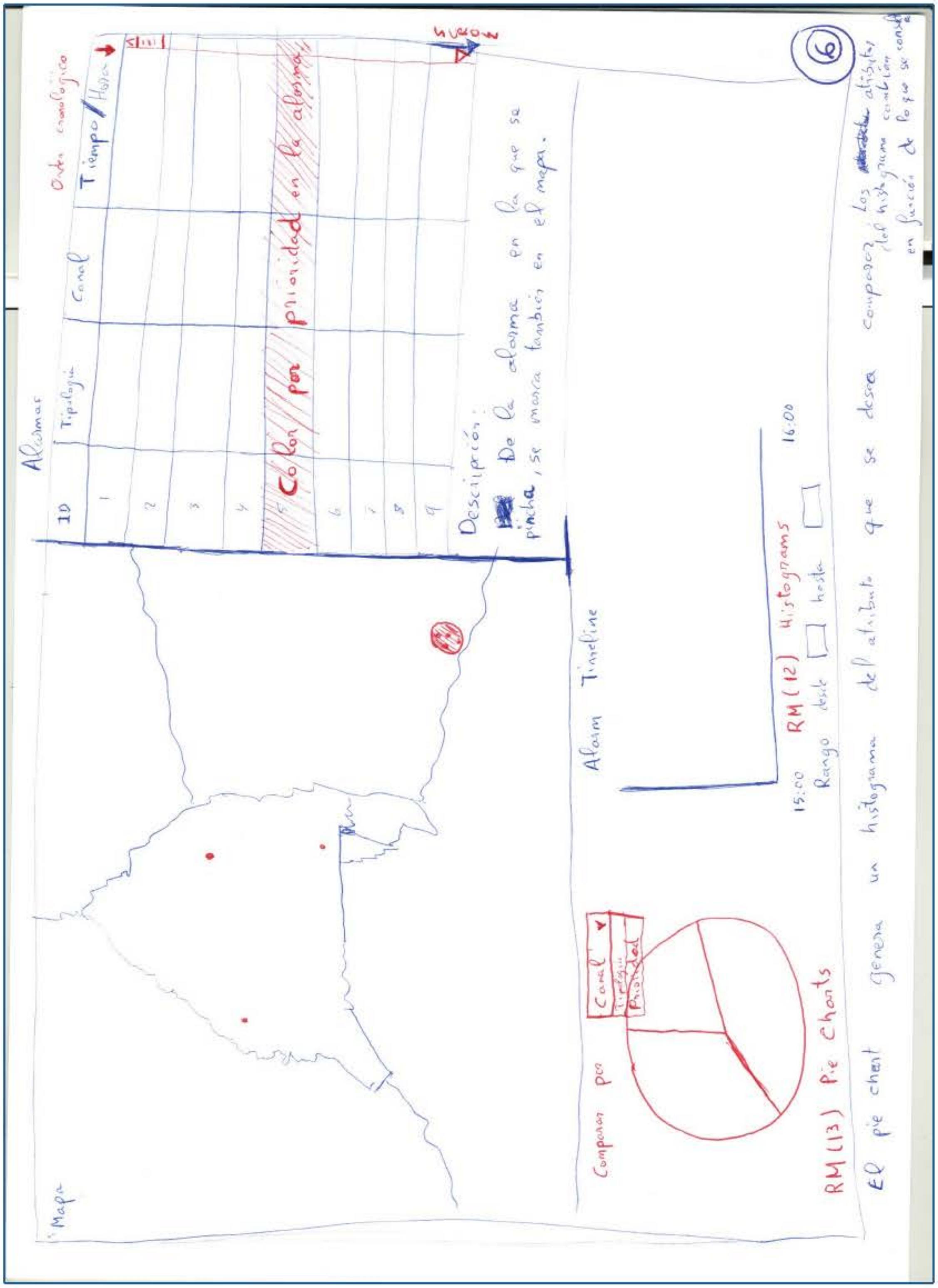
INTERACCIÓN

- IF(2) Display resolution management. Para manejar gran cantidad de info, mecanismos como scrolling.
- IF(1) Direct manipulation. Para la rapidez de uso de la interfaz
- IM(4) Brushing and linking. Las diferentes pantallas están coordinadas.
- IM(5) Zooming. En el mapa → más info.
- IM(6) Panning. En el mapa.
- IM(7) Scrolling. En el timeline.

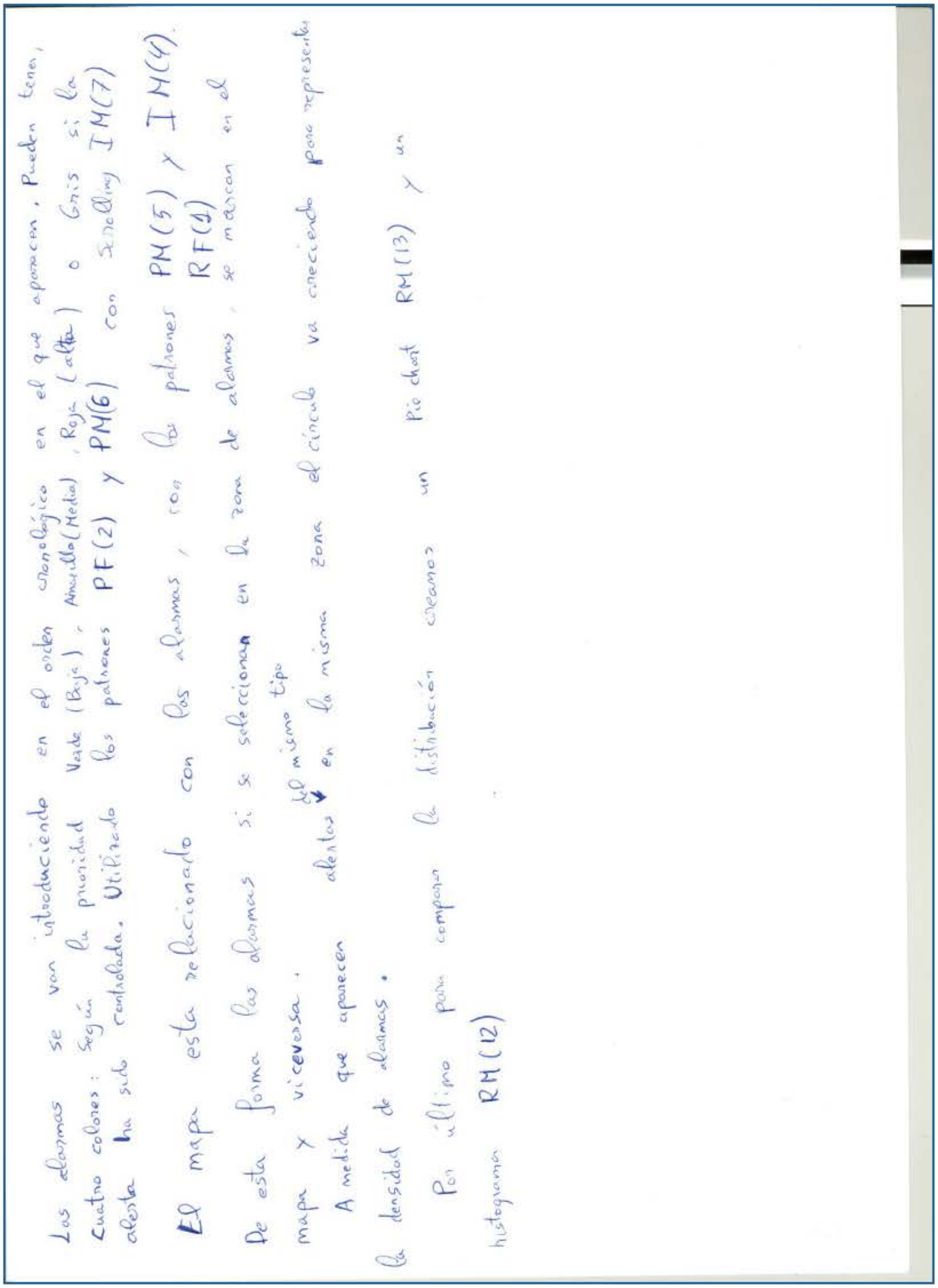
Annex Fig. B-8 Description of the design patterns applied to create the sketch by participant 4



Annex Fig. B-9 Sketch generated by participant 5 using the design pattern language



Annex Fig. B-10 Sketch generated by participant 6 using the design pattern language



Las alarmas se van introduciendo en el orden cronológico en el que aparecen, Pueden tener cuatro colores: Según la prioridad Verde (Baja), Amarillo (Media), Roja (alta) o Gris si la alerta ha sido controlada. Utilizado los patrones PF(2) y PM(6) con Scrolling IM(7)

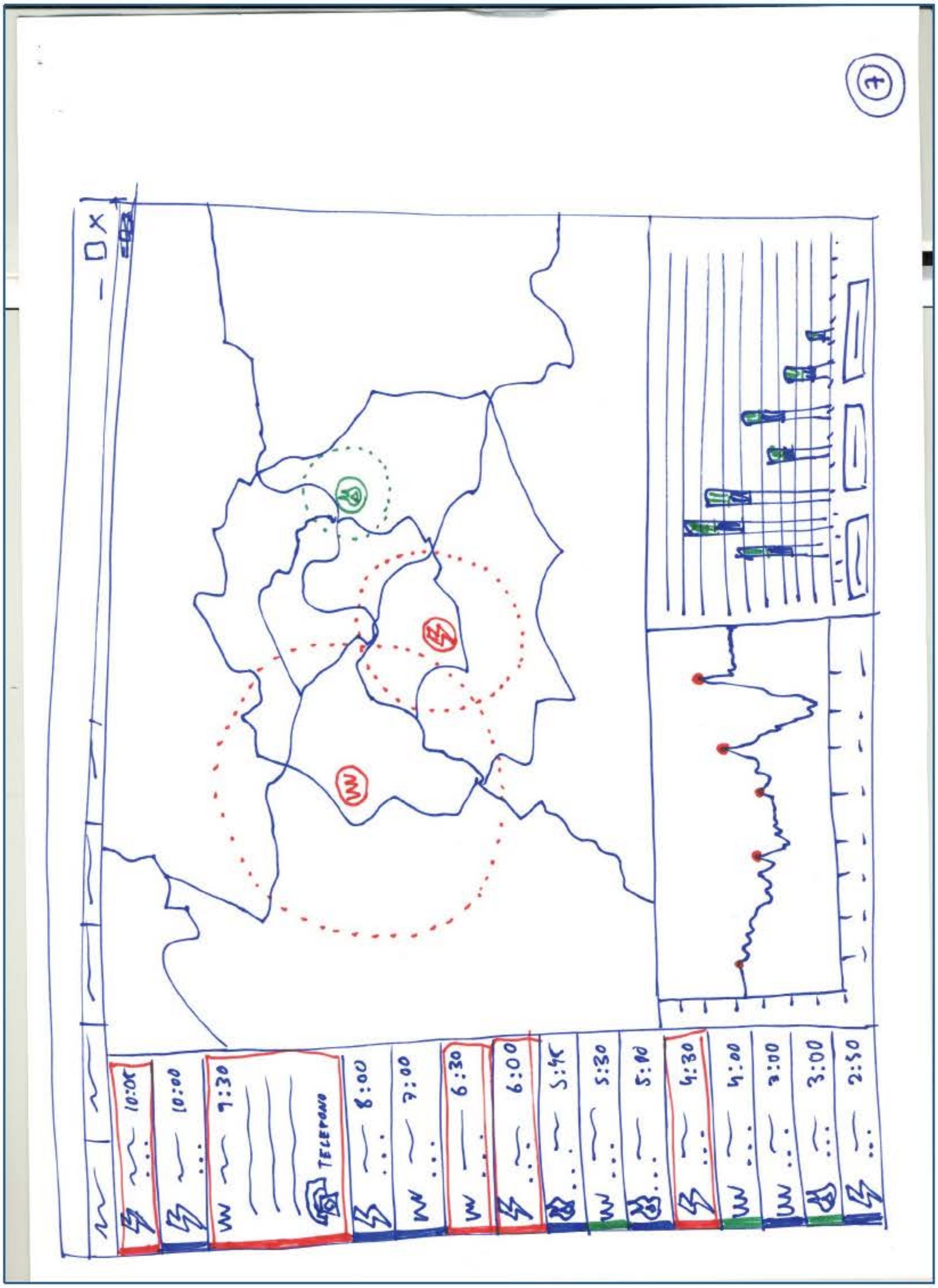
El mapa esta relacionado con las alarmas, con los patrones PM(5) y IM(4). RF(1)

De esta forma las alarmas si se seleccionan en la zona de alarmas, se marcan en el mapa y viceversa.

A medida que aparecen alertas del mismo tipo en la misma zona el circulo va creciendo para representar la densidad de alarmas.

Por último para comparar la distribución creamos un Pie chart RM(13) y un histograma RM(12)

Annex Fig. B-11 Description of patterns applied to create the sketch by participant 6



Annex Fig. B-12 Sketch generated by participant 7 using the design pattern language

RF(5) LISTS: Para mostrar una lista de alarmas en el orden que han ocurrido.

RF(1) HIGHLIGHTNING: En un mapa para ver donde han ocurrido las alarmas y a la zona a la que afectan.

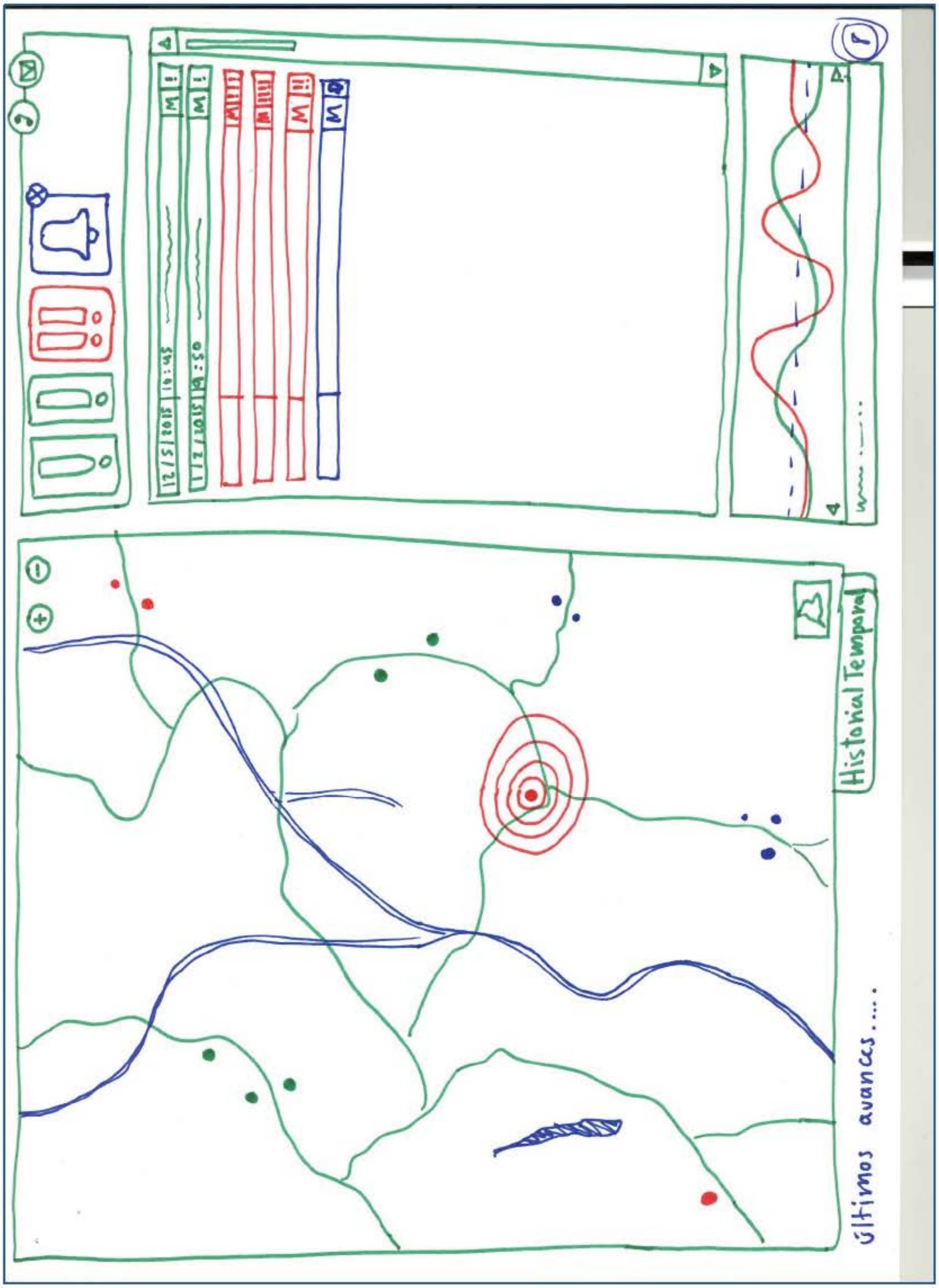
RF(4) y RM(12): Para comparar la distribución de alarmas por canal, tipología y prioridad.

RM(7) FLASHING: Para distinguir las alarmas con mayor prioridad.

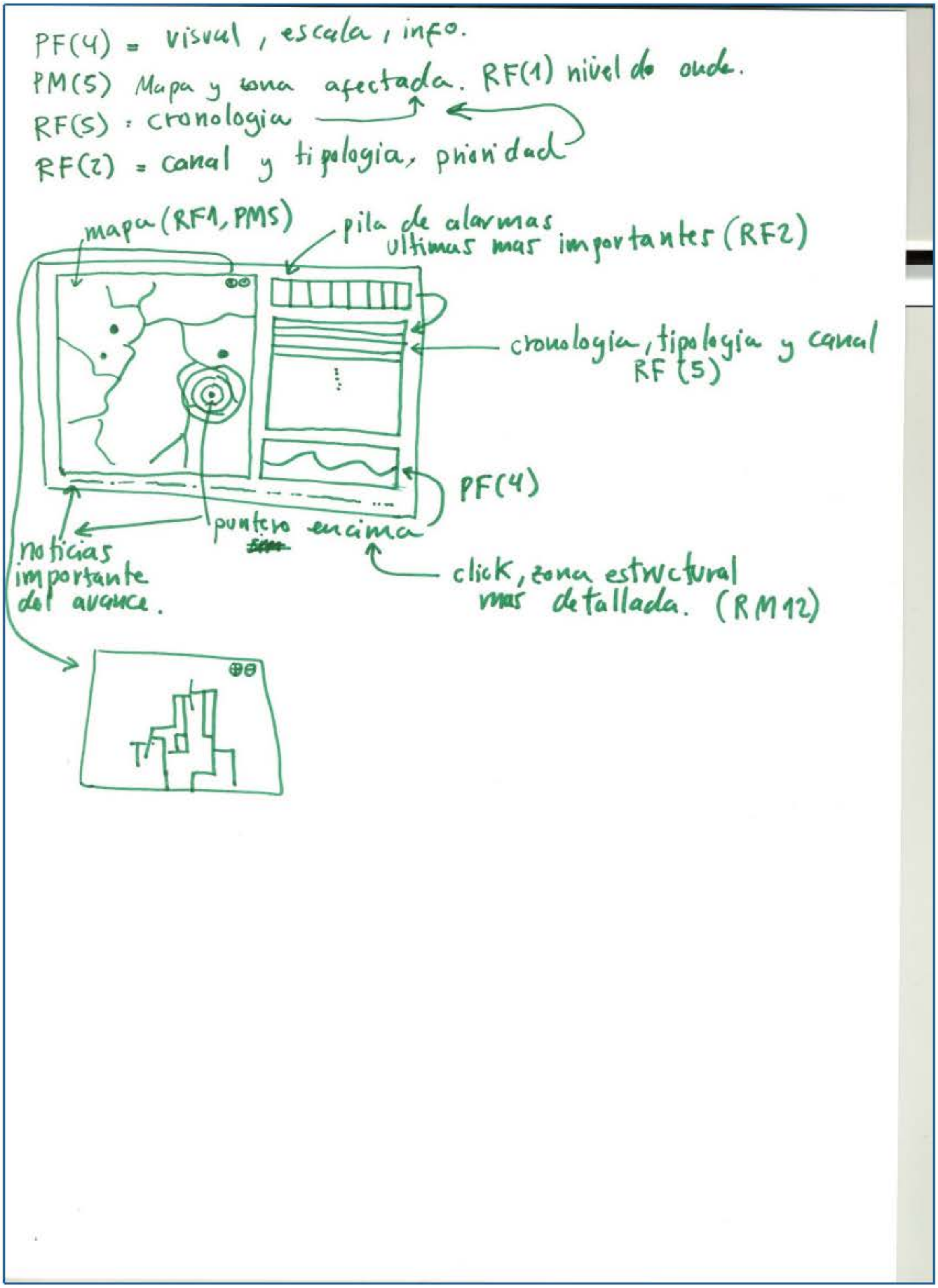
IM(3) DYNAMIC QUERIES: Para ver y seleccionar elementos de forma dinámica entre la lista y el mapa.

IM(7) SCROLLING: para ver la lista de alarmas completa porque puede extenderse demasiado.

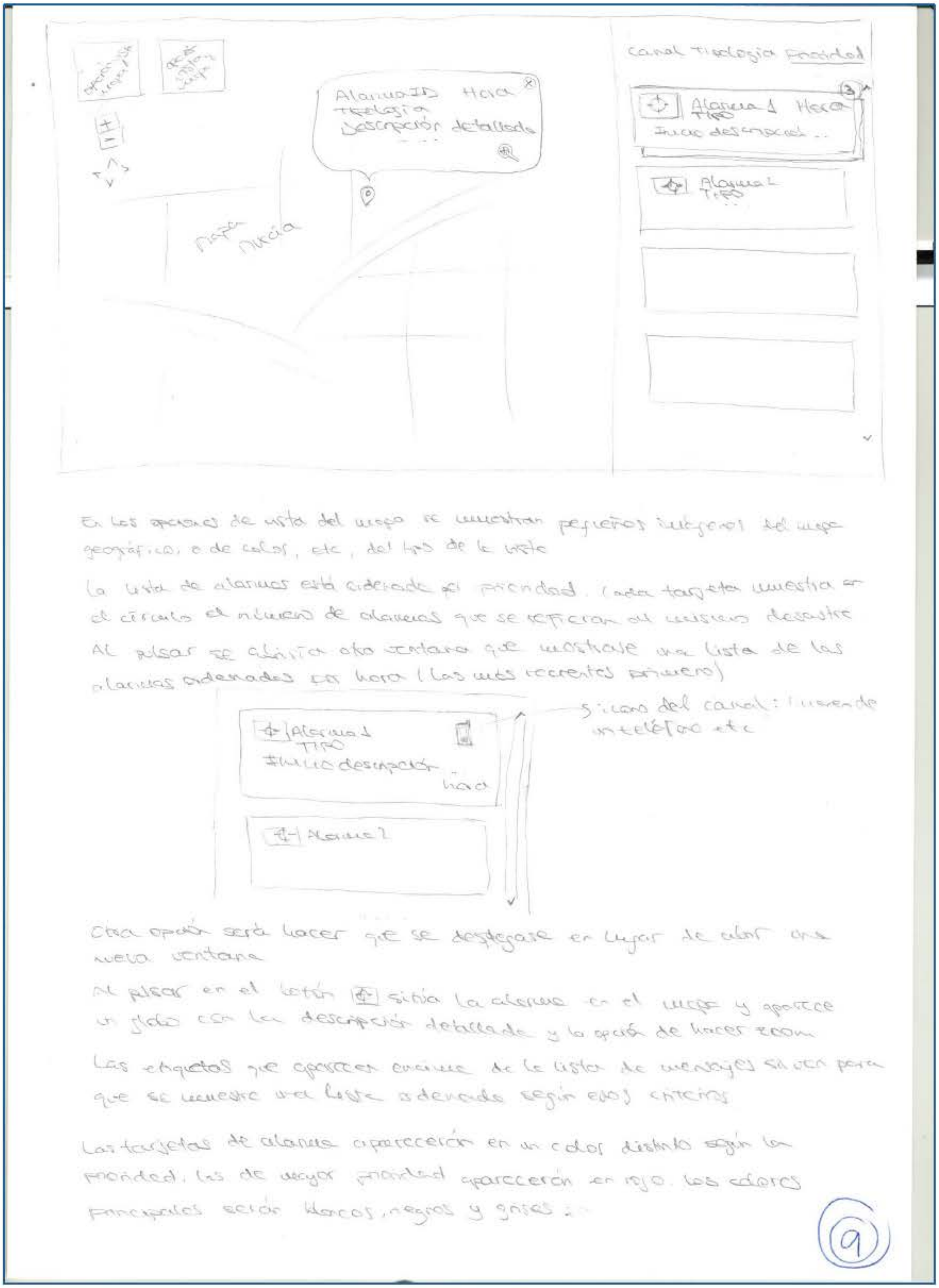
Annex Fig. B-13 Description of patterns applied to create the sketch by participant 7



Annex Fig. B-14 Sketch generated by participant 8 using the design pattern language



Annex Fig. B-15 Description of patterns applied to create the sketch by participant 8



Annex Fig. B-16 Sketch generated by participant 9 using the design pattern language

Patrones utilizados:

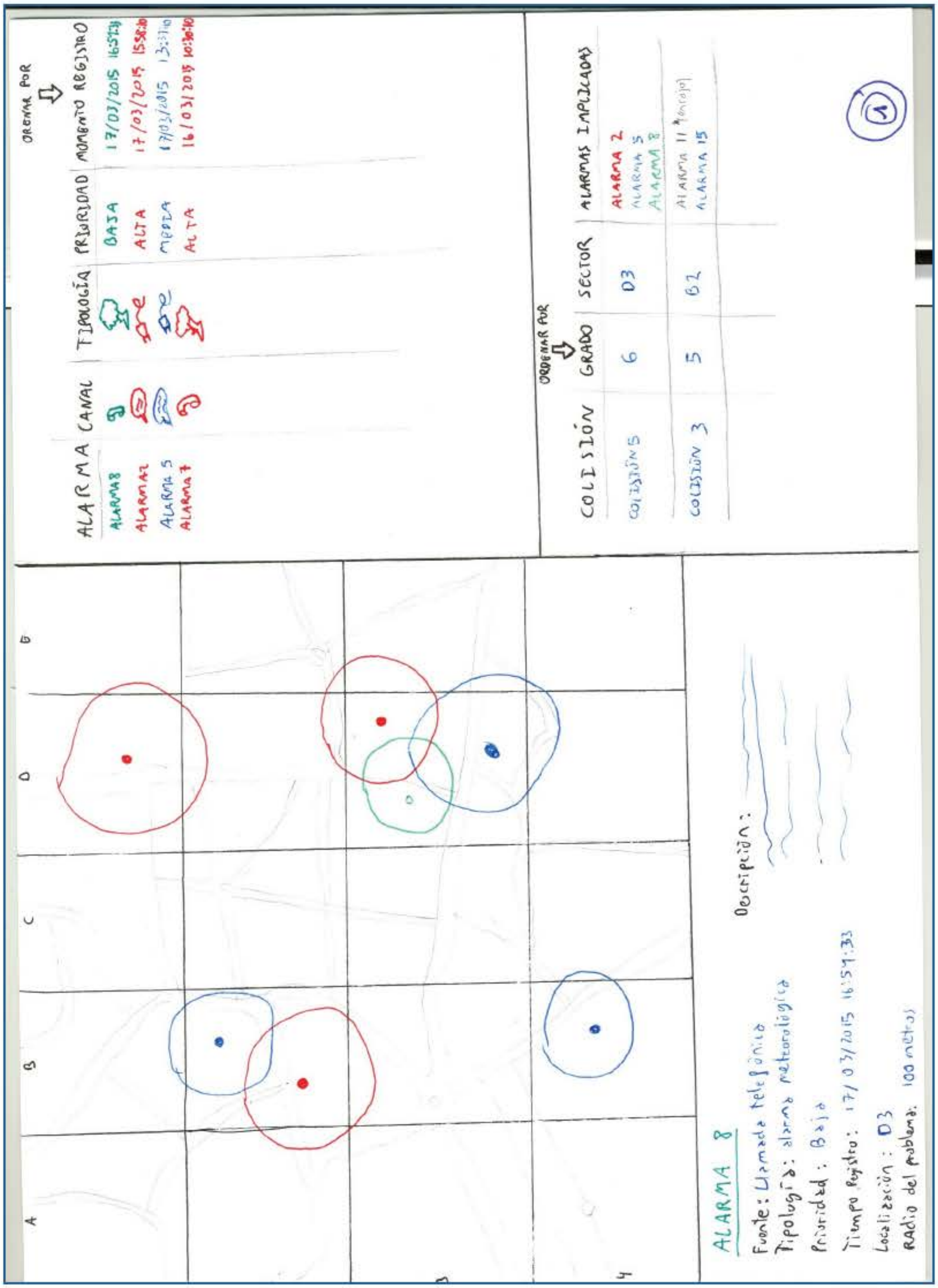
PF(3): En las tarjetas sale poca información pero al pulsar salen el resto de tarjetas relacionadas y al pulsar en la localización una descripción más detallada sobre el mapa

RF(3): Es fácil distinguir la naturaleza de los alarmas y relacionales

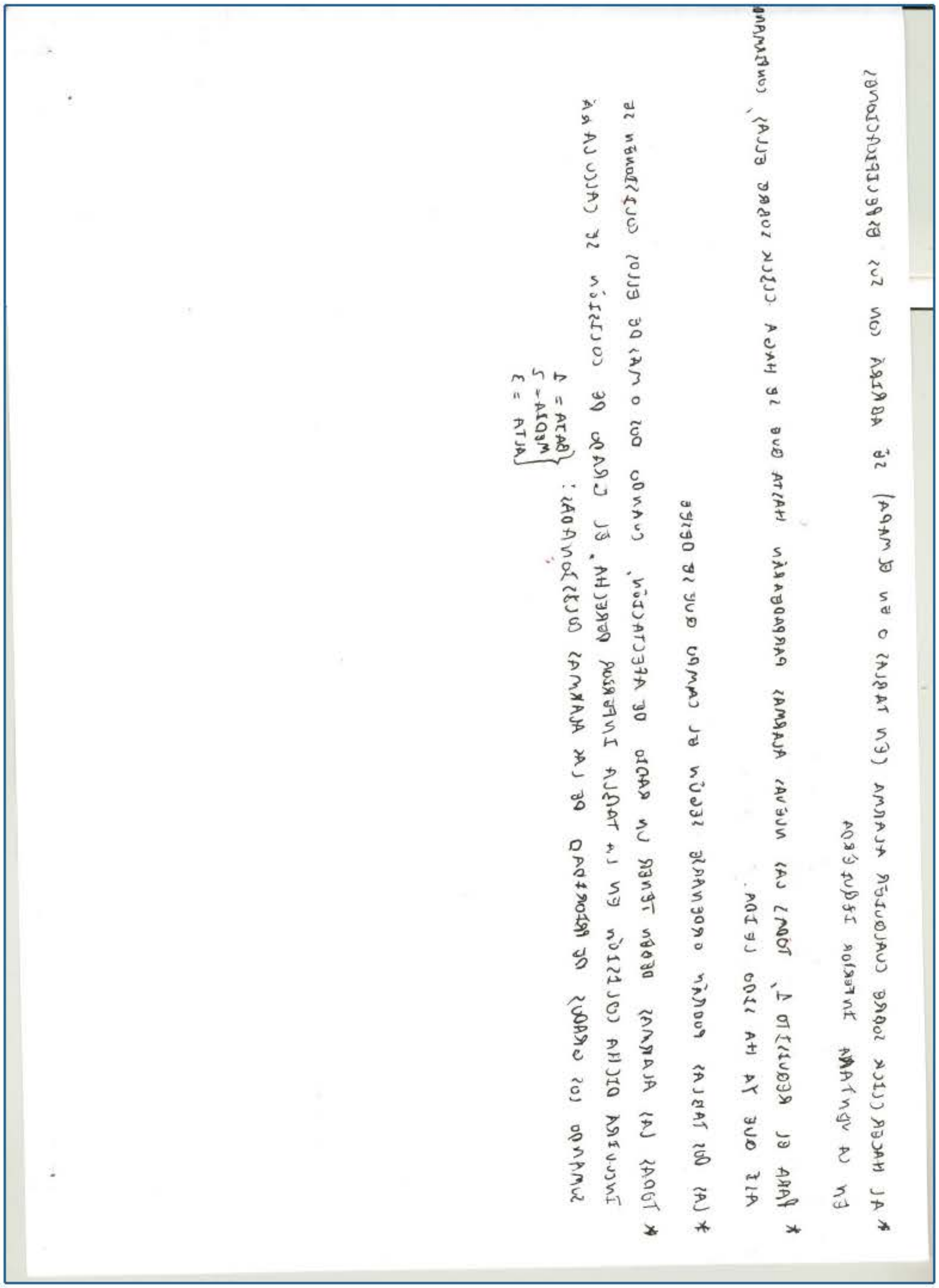
RF(5): Se usa una vista que puede ordenarse por canal, tipología o prioridad.

In(4): Links
In(7): Scrolling
En(9): mapa para las localizaciones
In(5): Zoom

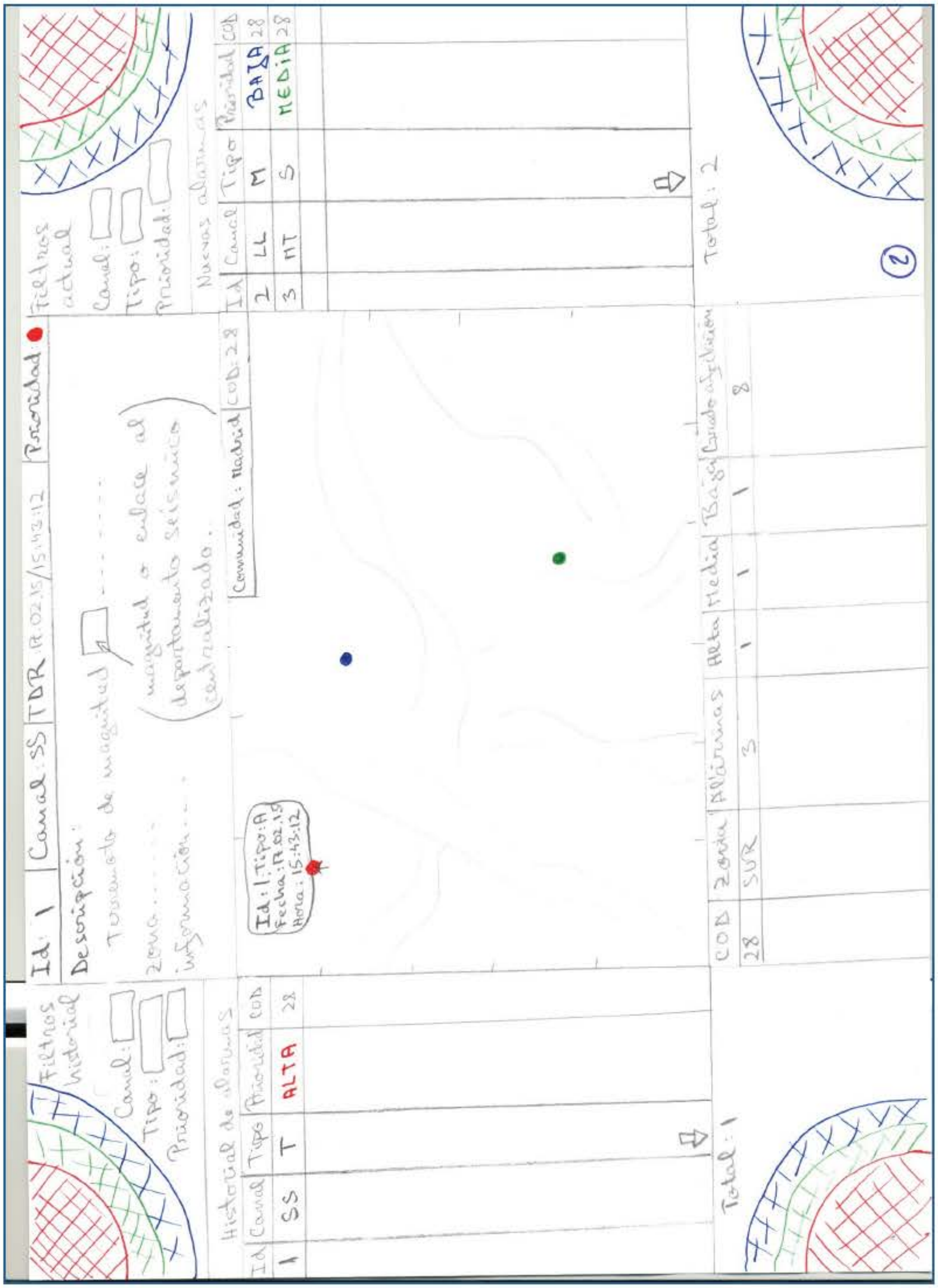
Annex Fig. B-17 Description of the patterns applied to create the sketch by participant 9



Annex Fig. B-18 Sketch generated by participant 10 without using design patterns



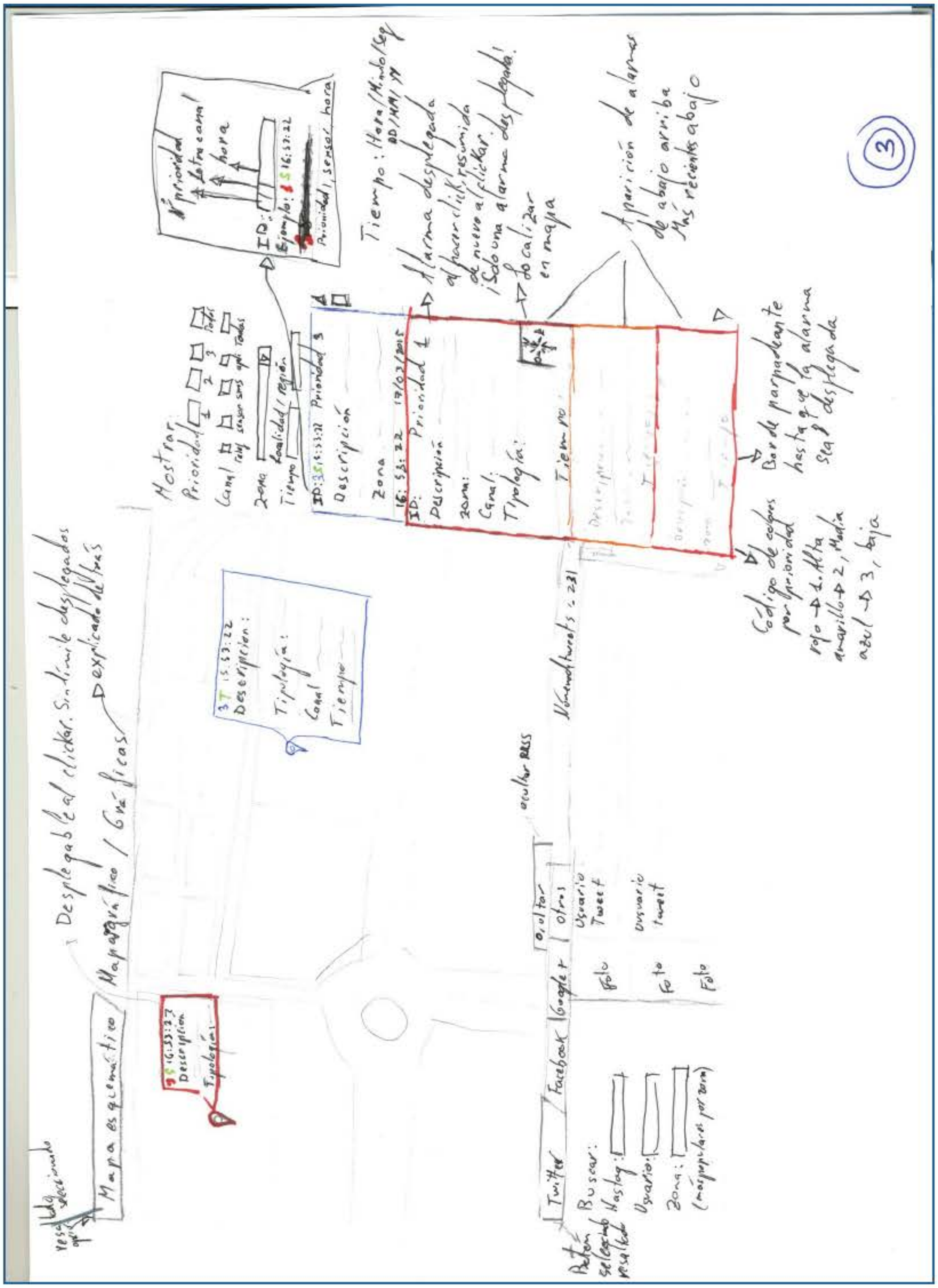
Annex Fig. B-19 Description of the sketch created by participant 10



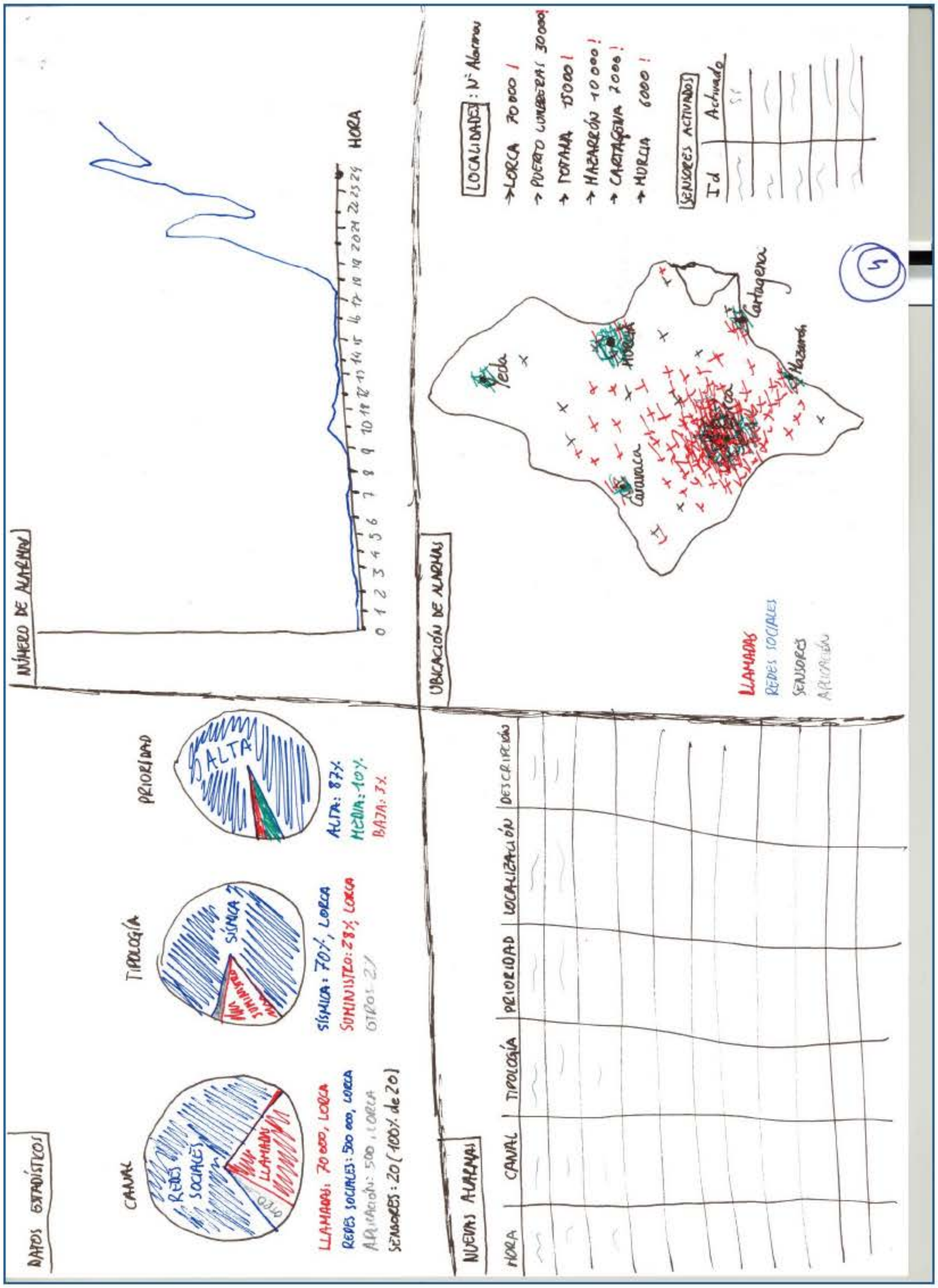
Annex Fig. B-20 Sketch generated by participant 11 without using the design pattern language

He elegido este layout debido a la importancia de la información que aparece en esta interfaz.
Con respecto al funcionamiento las nueva alarmas, al ser verificadas (pinchadas), pasan al historial de alarmas y se localizan en el mapa. A la vez, arriba nos aparece la descripción del alarma. Las esquinas nos señala el tipo de alarma con una codificación de colores, y no se apagan hasta que la alarma que han generado dicha señalización, no se verifica en la parte de las nuevas alarmas. Tanto las alarmas nuevas, como las antiguas, pueden ser filtradas por el canal, el tipo, y la prioridad. En la parte de abajo tenemos datos estadísticos sobre las zonas de las distintas comunidades afectadas, con un valor de afectación según los tipos y el número de alarmas que afectan cada zona, siendo 1 el valor de más afectada.

Annex Fig. B-21 Description of the sketch created by participant 11



Annex Fig. B-22 Sketch created by participant 12 without using the design pattern language



Annex Fig. B-23 Sketch created by participant 13 using the design pattern language

COMENTARIOS AL DISEÑO

Se ha dividido la interfaz en cuatro cuadrantes, para poder visualizar el máximo de información posible sin tener que interaccionar con ella, y poder reaccionar rápidamente.

→ DATOS ESTADÍSTICOS: proporciona datos sobre el total de alarmas recibidas en el día, atendiendo a su tipología, canal y prioridad.

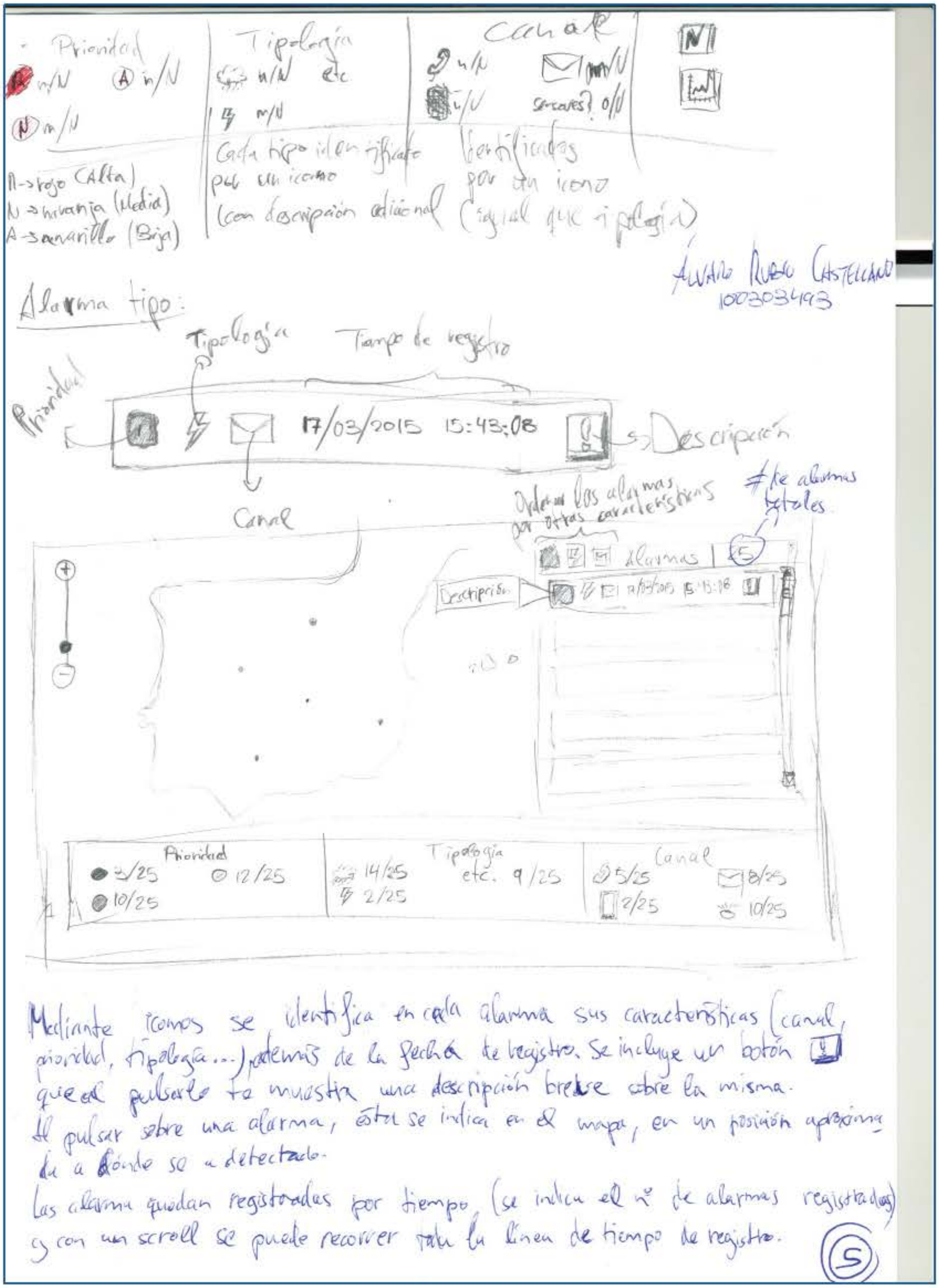
→ NÚMERO DE ALARMAS: relaciona las alarmas recibidas con las horas en que fueron recibidas.

→ NUEVAS ALARMAS: tabla detallada de las últimas alarmas recibidas.

→ UBICACIÓN DE ALARMAS: permite visualizar rápidamente las alarmas en su ubicación en la Comunidad de Murcia. Se usa un código de colores según la fuente:

- ROJO = llamadas
- AZUL = menciones en redes sociales
- NEGRO = sensores activados (se detalla el porcentaje en los datos estadísticos)
- GRIS = alertas enviadas por la aplicación.

Annex Fig. B-24 Description of the sketch created by participant 13

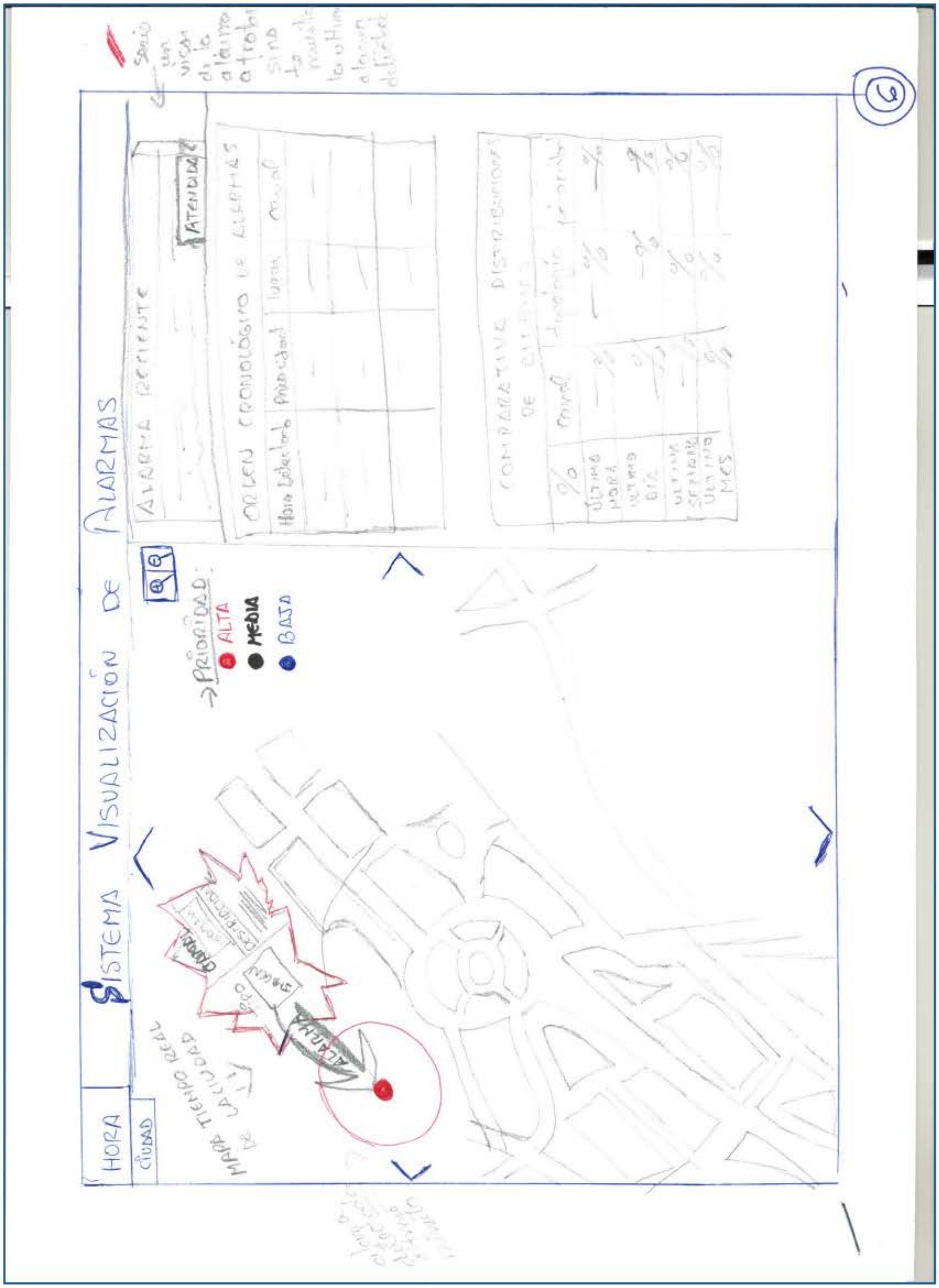


Annex Fig. B-25 Sketch generated by participant 14 without using the design pattern language

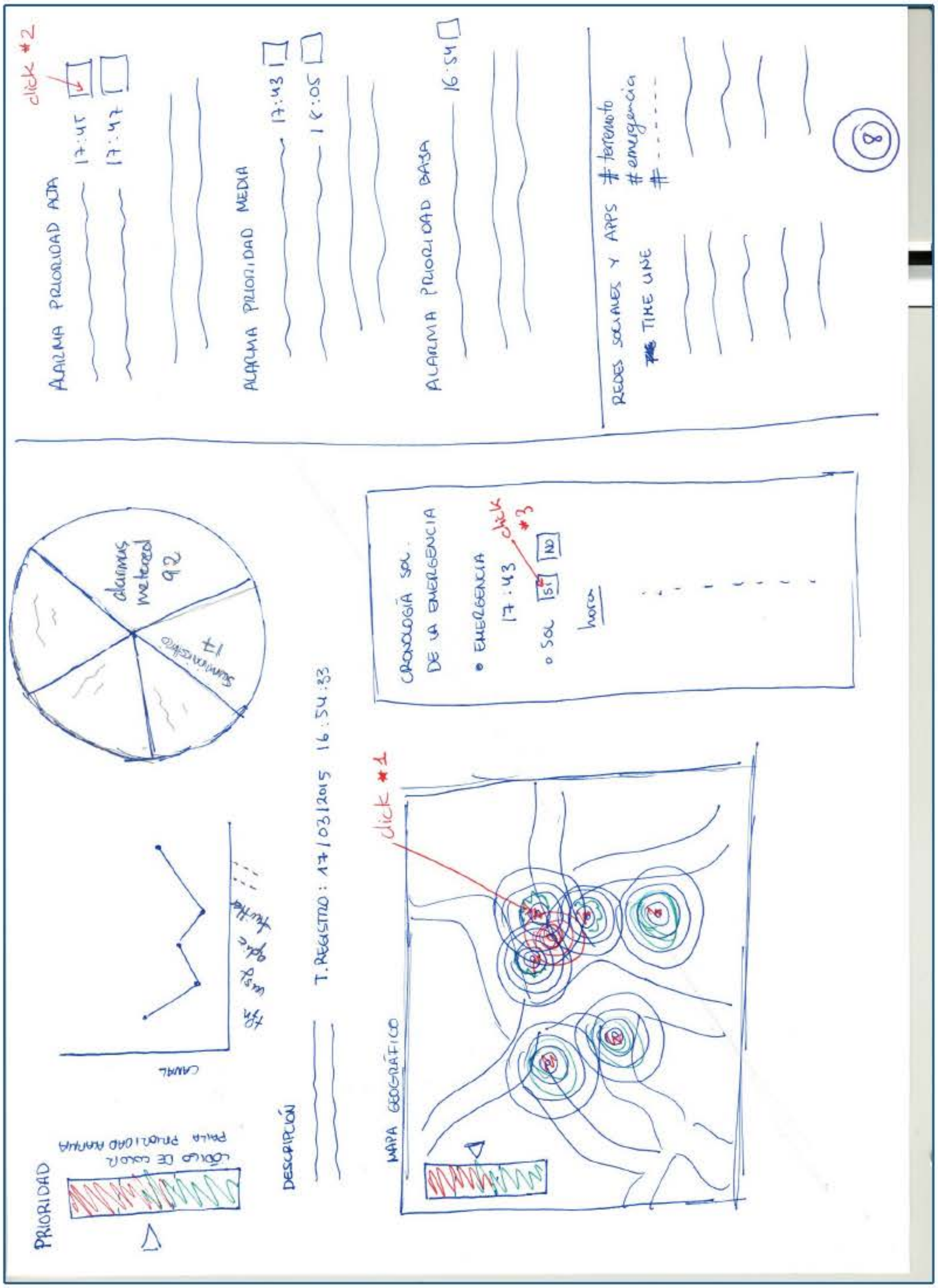
El mapa tiene un zoom ~~para poder~~ y permite el desplazamiento por el mismo.
En la parte inferior se representan ~~las dif~~ tres características principales de las alarmas (prioridad, tipología, canal) y una comparación del número de alarmas por cada una (los valores posibles de cada característica) con respecto del nº de alarmas totales.

En la línea de tiempo, se pueden ordenar las alarmas, a través de botones, por otras características que no sean la fecha.

Annex Fig. B-26 Description of the sketch created by participant 14



Annex Fig. B-27 Sketch created by participant 15 without using the design pattern language



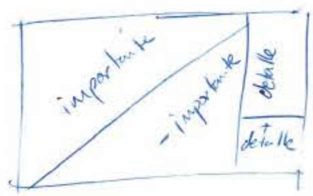
Annex Fig. B-28 Sketch created by participant 16 without using the design pattern language

CLICK 1: CAMBIAMOS A ESA ALARMA EN ESA SIT. GEOGRÁFICA

CLICK 2: ESTABLECEMOS EN LA PARTE IZQ LA NUEVA ALARMA

click 3: COMUNICAMOS SOBRE LA ASISTENCIA CIVIL Y SI ESTÁ SOL YA

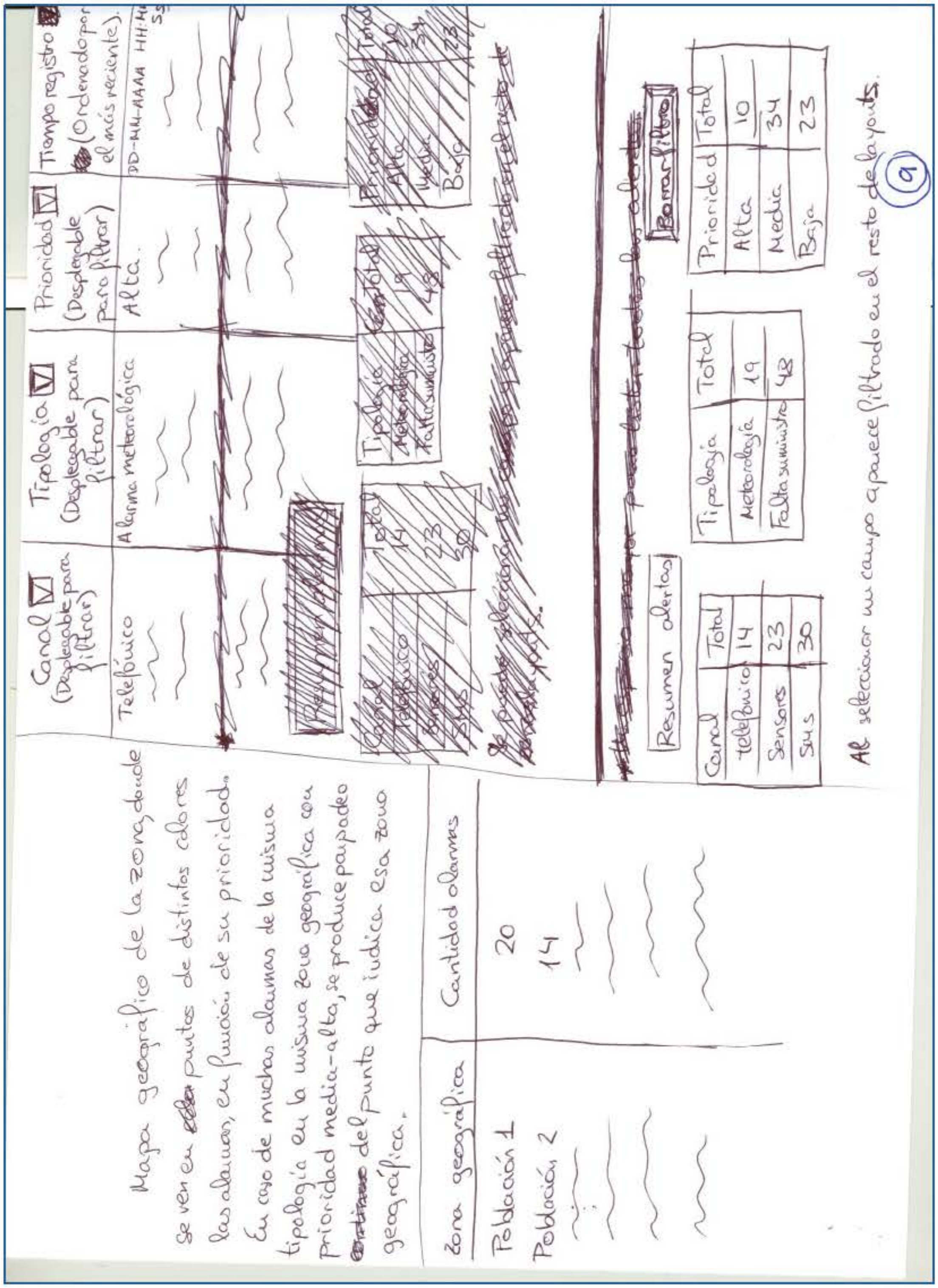
LA PRIORIDAD SIGUE UN CÓDIGO DE COLORES ADEMÁS AQUELLAS PARTES IMPORTANTES SE SITÚAN A LA IZQUIERDA Y EL DETALLE A LA DERECHA



EL CÓDIGO DE COLORES SEGUIDO SERÍA UNA ESCALA DE AMARILLO-ROJO

HE DECIDIDO PONER POCA INTERACCIÓN Y LA MÁXIMA VISUALIZACIÓN DE LOS DATOS IMPORTANTES PARA QUE NO SE PIERDA INFORMACIÓN EN EL CAMINO.

Annex Fig. B-29 Description of the sketch created by participant 16

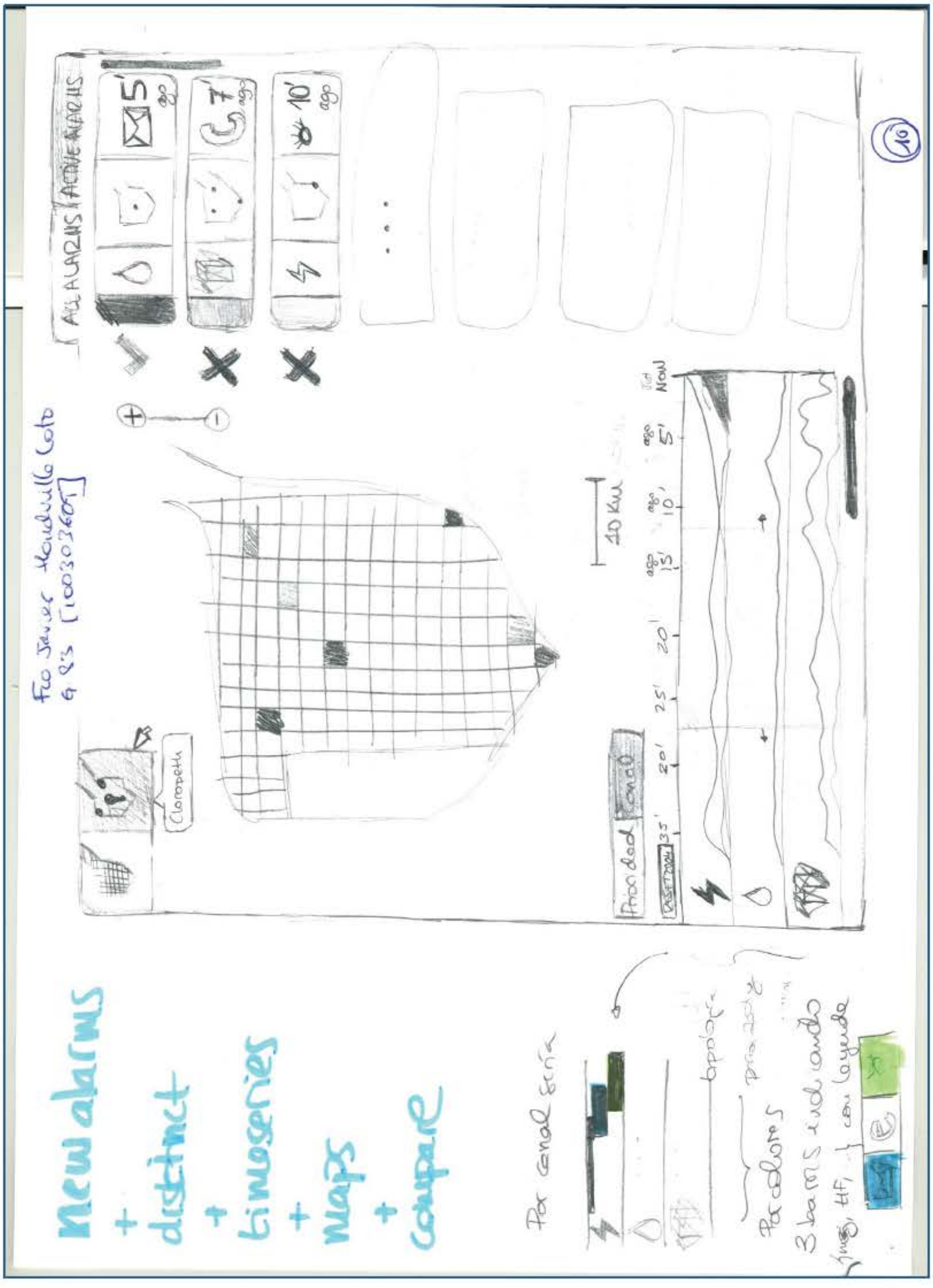


Annex Fig. B-30 Sketch created by participant 17 without using the design pattern language

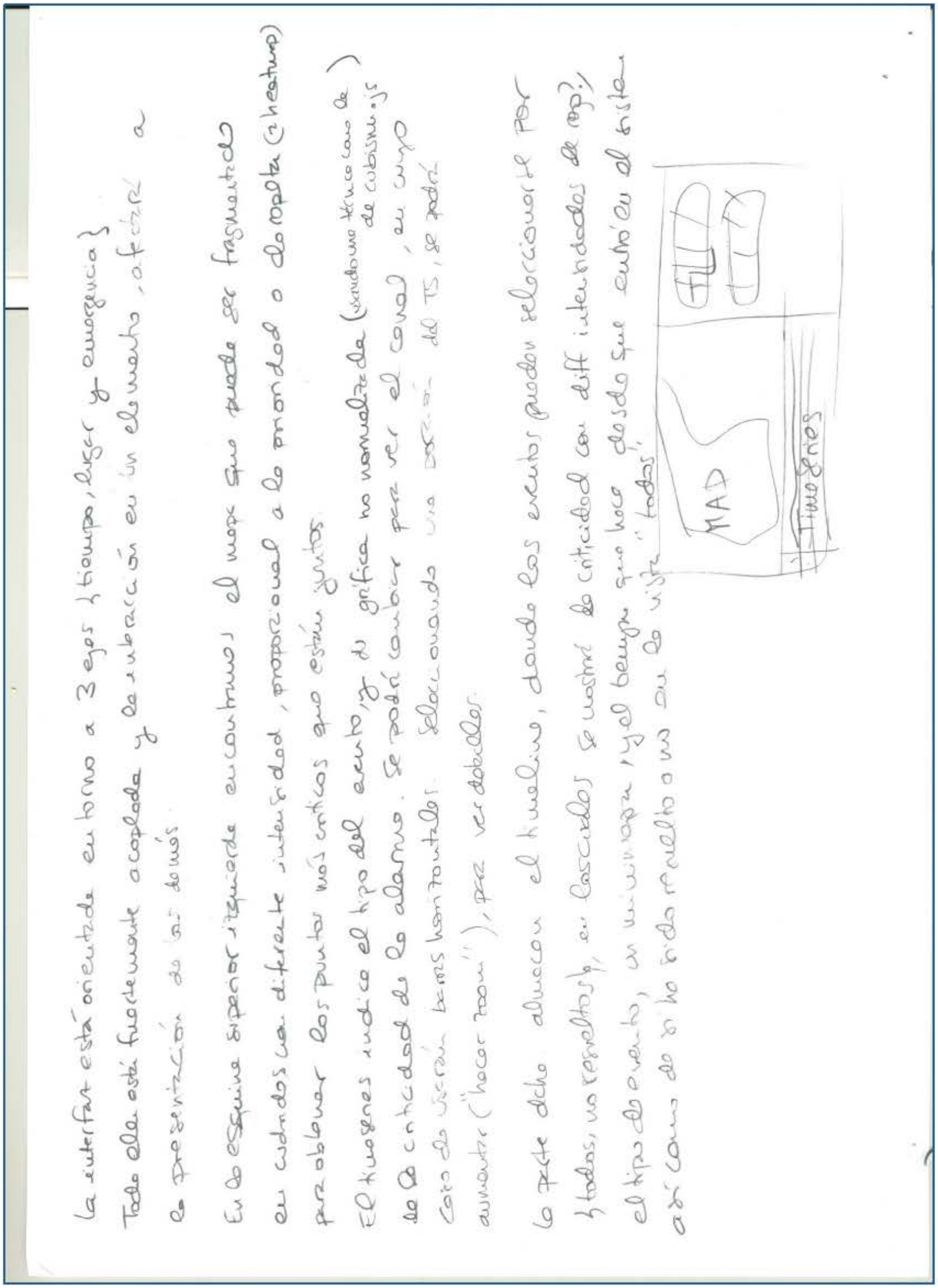
La interfaz se distingue en dos zonas; la parte izquierda representa las características geográficas de la alarma, tanto de forma visual con un mapa, como en forma tabular con una tabla.

La parte derecha aparece el listado general de alarmas, con la prioridad tipología y canal con un desplegable. También aparece el campo tiempo registro que ordena las alarmas. No me ha quedado espacio, pero habría otra columna más con la descripción de la alarma.
El desplegable se utiliza para filtrar por cada aspecto de la alarma.
En la parte inferior derecha hay un resumen de las alarmas con diferentes clasificaciones. Al filtrar por uno de estos campos aparece el resto de layouts filtrados.

Annex Fig. B-31 Description of the sketch created by participant 17



Annex Fig. B-32 Sketch created by participant 18 without using the design pattern language



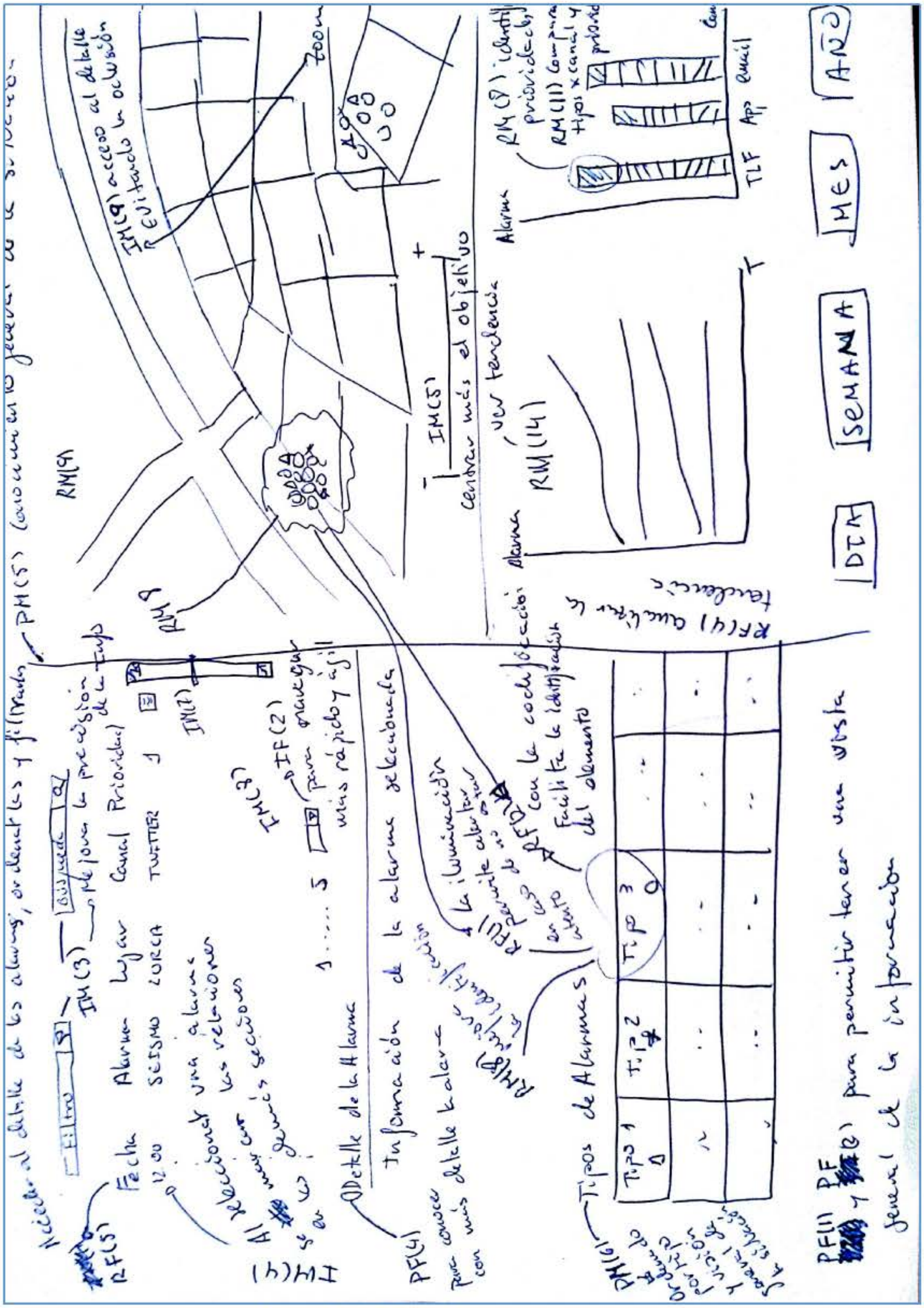
La interfaz está orientada en torno a 3 ejes {tiempo, lugar y emergencia}
Todo ello está fuertemente acoplada y la interacción en un elemento, afectará a la presentación de los demás.

En la esquina superior izquierda encontramos el mapa que puede ser fragmentado en cuadros con diferente intensidad, proporcional a la prioridad o choropleta (o heatmap) para obtener los puntos más críticos que están juntos.
El timeseries indica el tipo del evento, y la gráfica no normalizada (usando una técnica como la de cubism.js) de la criticidad de la alarma. Se podrá cambiar para ver el canal, en cuyo caso se verán barras horizontales. Seleccionando una porción del TS, se podrá aumentar ("hacer zoom"), para ver detalles.

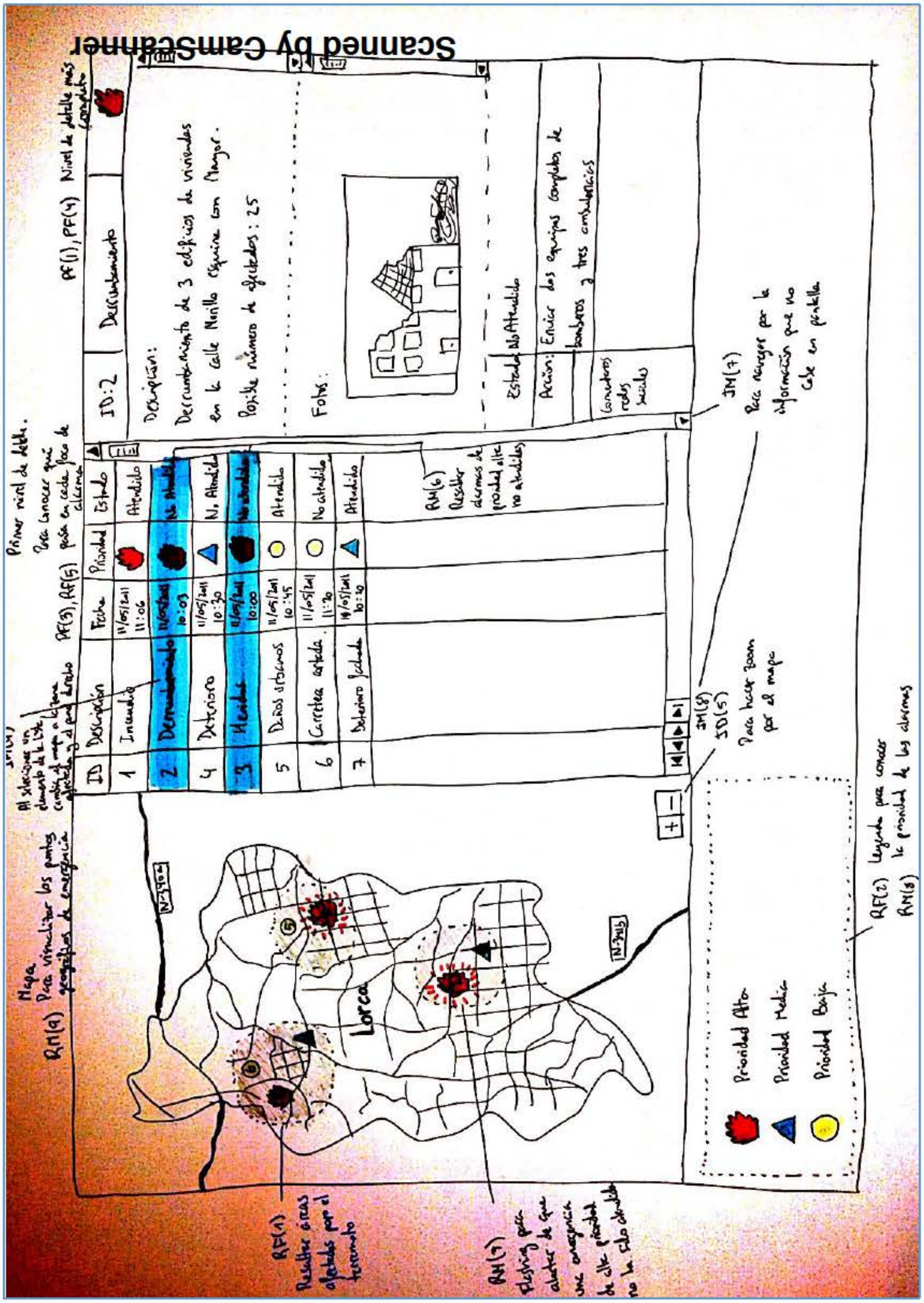
La parte dcha alberga el timeline, donde los eventos pueden seleccionarse por {todos, no resueltos}, en cascadas se mostrará la criticidad con diff intensidades de rojo?, el tipo de evento, un minimapa, y el tiempo que hace desde que entró en el sistema así como de si ha sido resuelto o no en la vista "todos".

Annex Fig. B-33 Description of the sketch created by participant 18

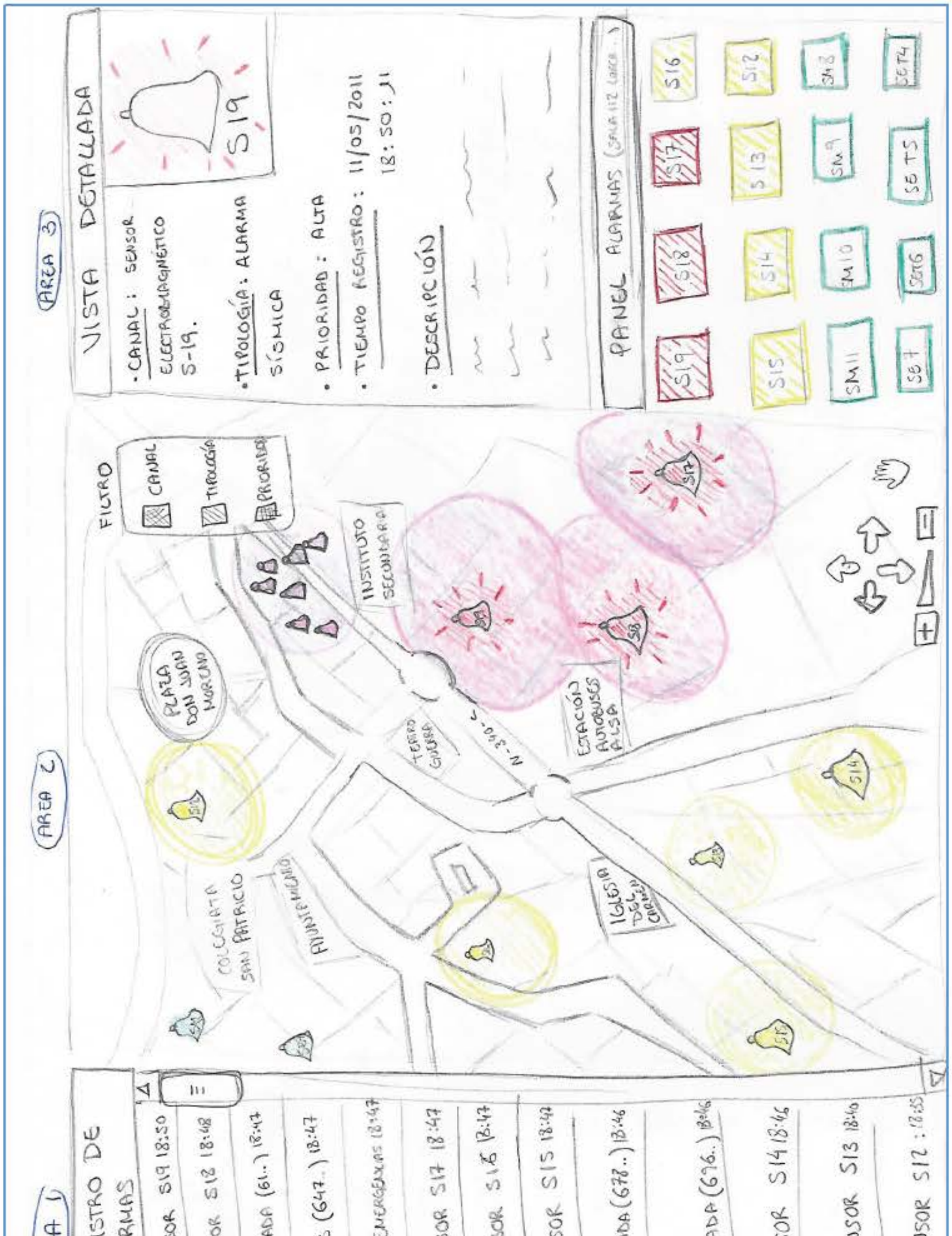
Annex C - Collection of sketches (Second round of the Experimental Evaluation)



Annex Fig. C-1 Sketch created by participant 1



Annex Fig. C-2 Sketch created by participant 2



Annex Fig. C-3 Sketch created by participant 3

PATRONES DE DISEÑO

→ PF(1) Details hierarchy

Aplicado en área 2 sobre el mapa, que representaría el primer nivel de la jerarquía. Se podría interactuar con los elementos del mapa e ir viendo una descripción detallada en el área 3.

→ PM(6) Tiled Layout

Como el área 2 (mapa) puede que no muestre todas las alarmas (por haberse hecho zoom, etc.) se incorpora un tiled layout en el área 3, "panel de alarmas", para que se puede ver de un vistazo todas las alarmas y su estado.

→ PM(5) Overview and Detail

Para no perder el contexto, tanto al interactuar con la lista del área 1, "registro de alarmas", como sobre los elementos del mapa del área 2, se despliega una vista detallada en el área 3 con la información relevante aso

→ RF(2) Visual Coding Scheme

Se utilizan propiedades visuales para ayudar a desific la prioridad de las alarmas.

→ RF(1) Highlighting

Se utiliza el highlighting para ayudar a establecer las relaciones espaciales entre alarmas y así poder identificar las áreas afectadas.

→ RF(7) Flashing

Se utiliza para destacar alarmas con prioridad máxima.

Annex Fig. C-4 First page of the description of the sketch created by participant 3

→ IM (5) Zooming

Se puede hacer zoom sobre el mapa.

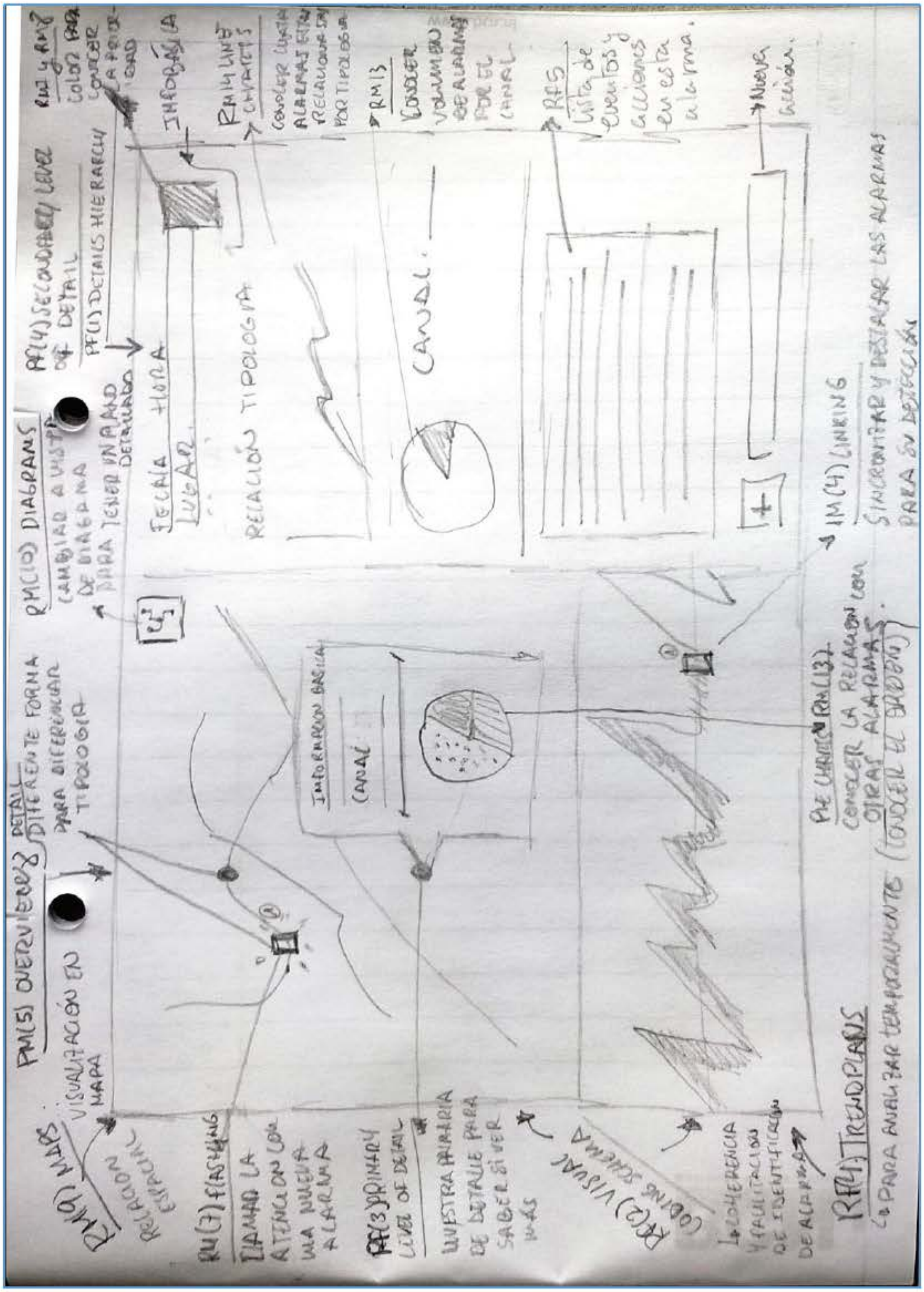
→ IM (6) Panning

Se puede hacer panning mediante flechas o con la herramienta de mano.

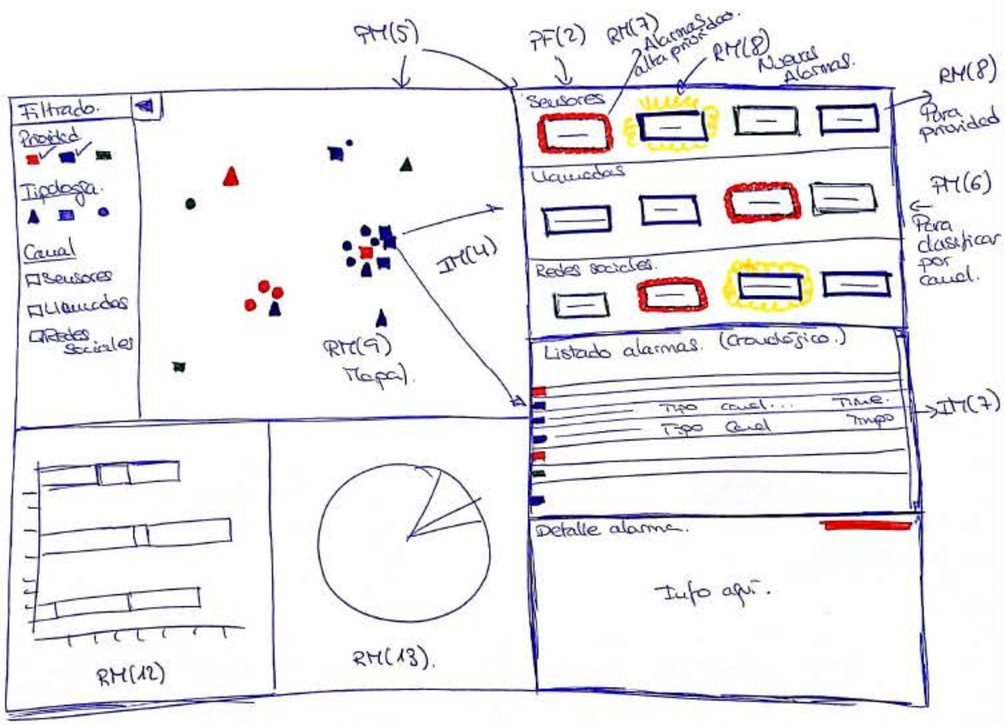
→ IM (7) Scrolling

En el área 1 se puede hacer scrolling para navegar entre el registro de alarmas. Las que se acaban de producir se colocan al principio (arriba), mientras que las más antiguas tienden a ir hacia el final (abajo).

Annex Fig. C-5 Second page of the description of the sketch created by participant 3



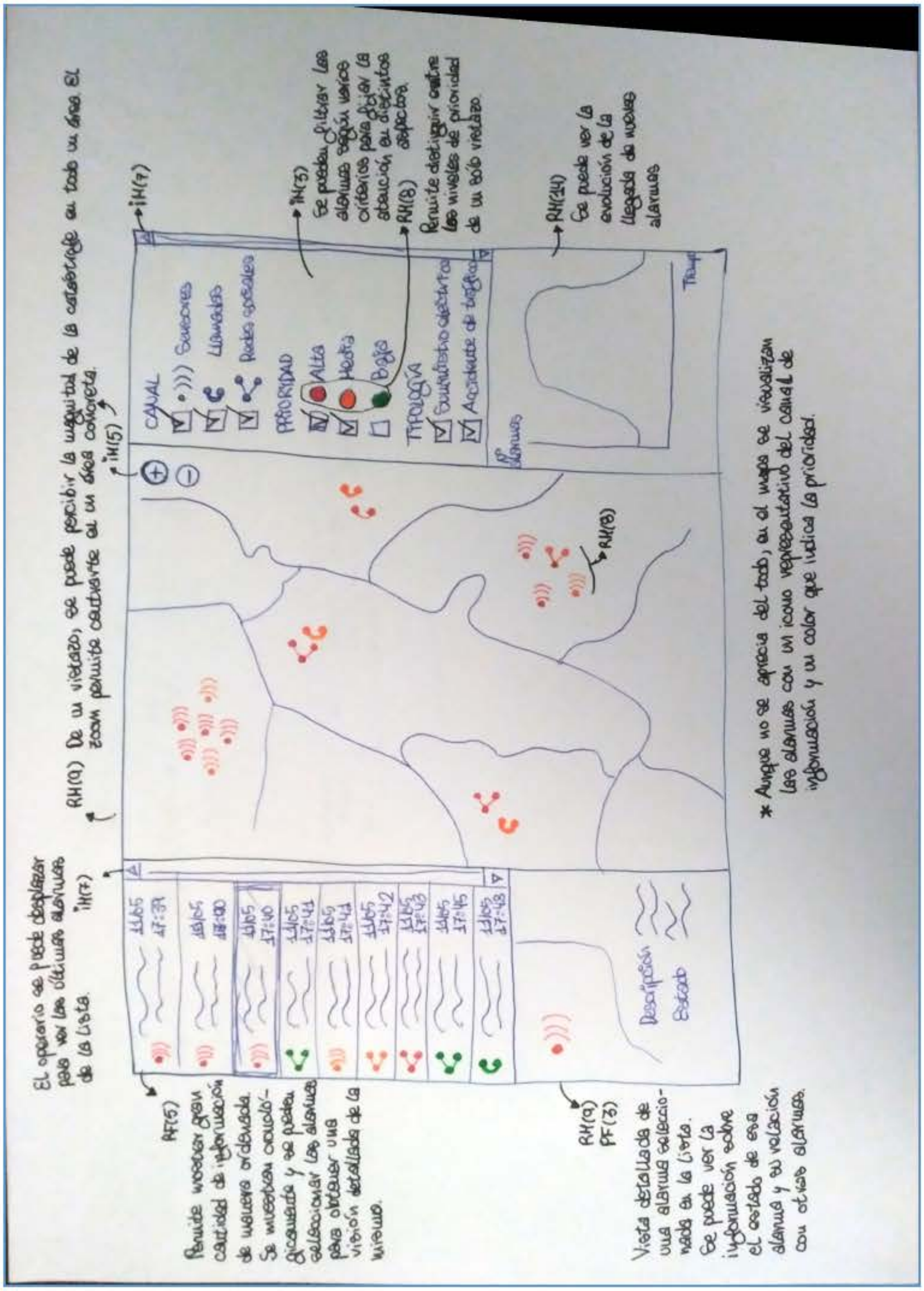
Annex Fig. C-6 Sketch created by participant 4



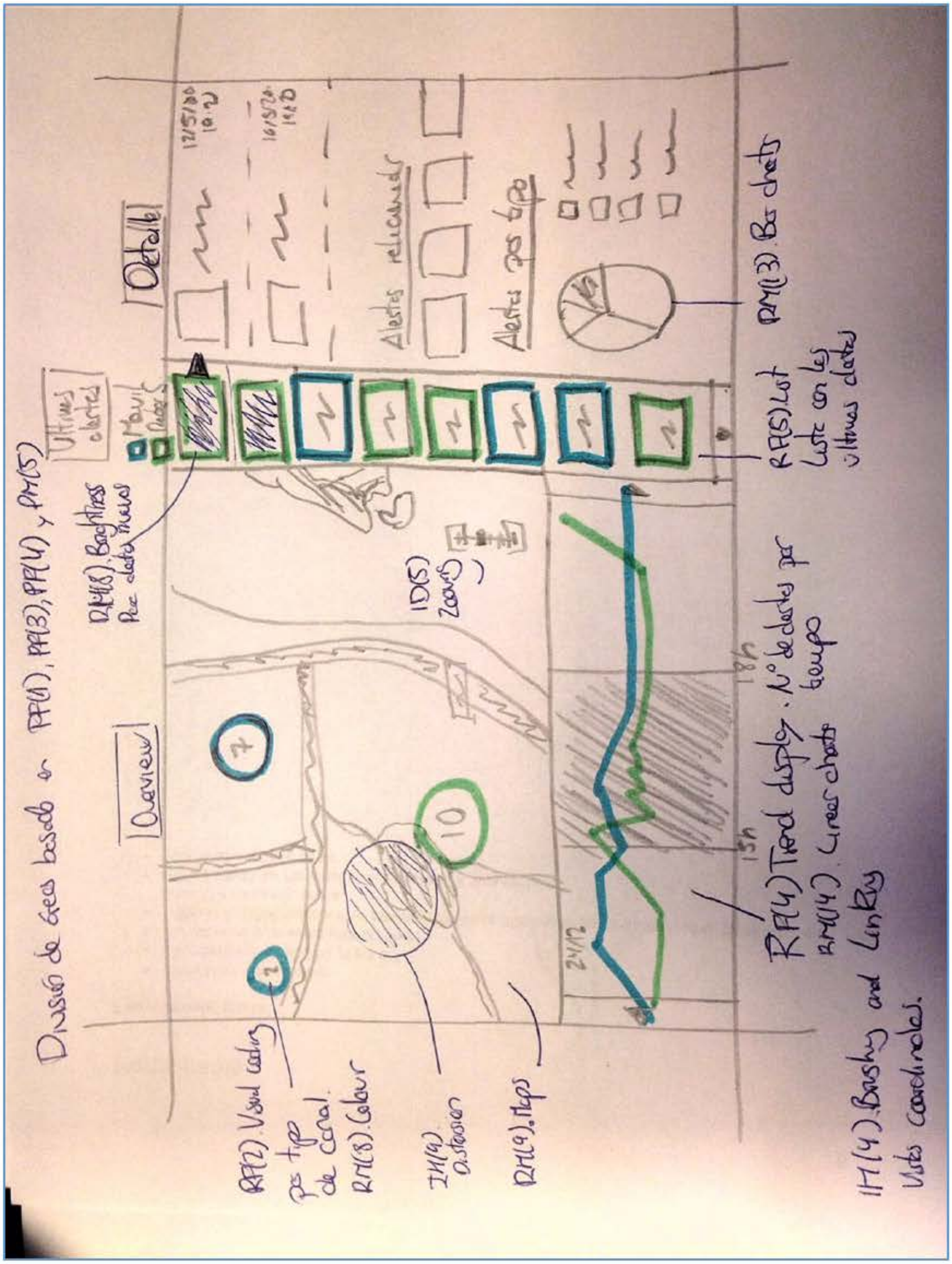
Scanned by CamScanner

Annex Fig. C-7 Sketch created by participant 5

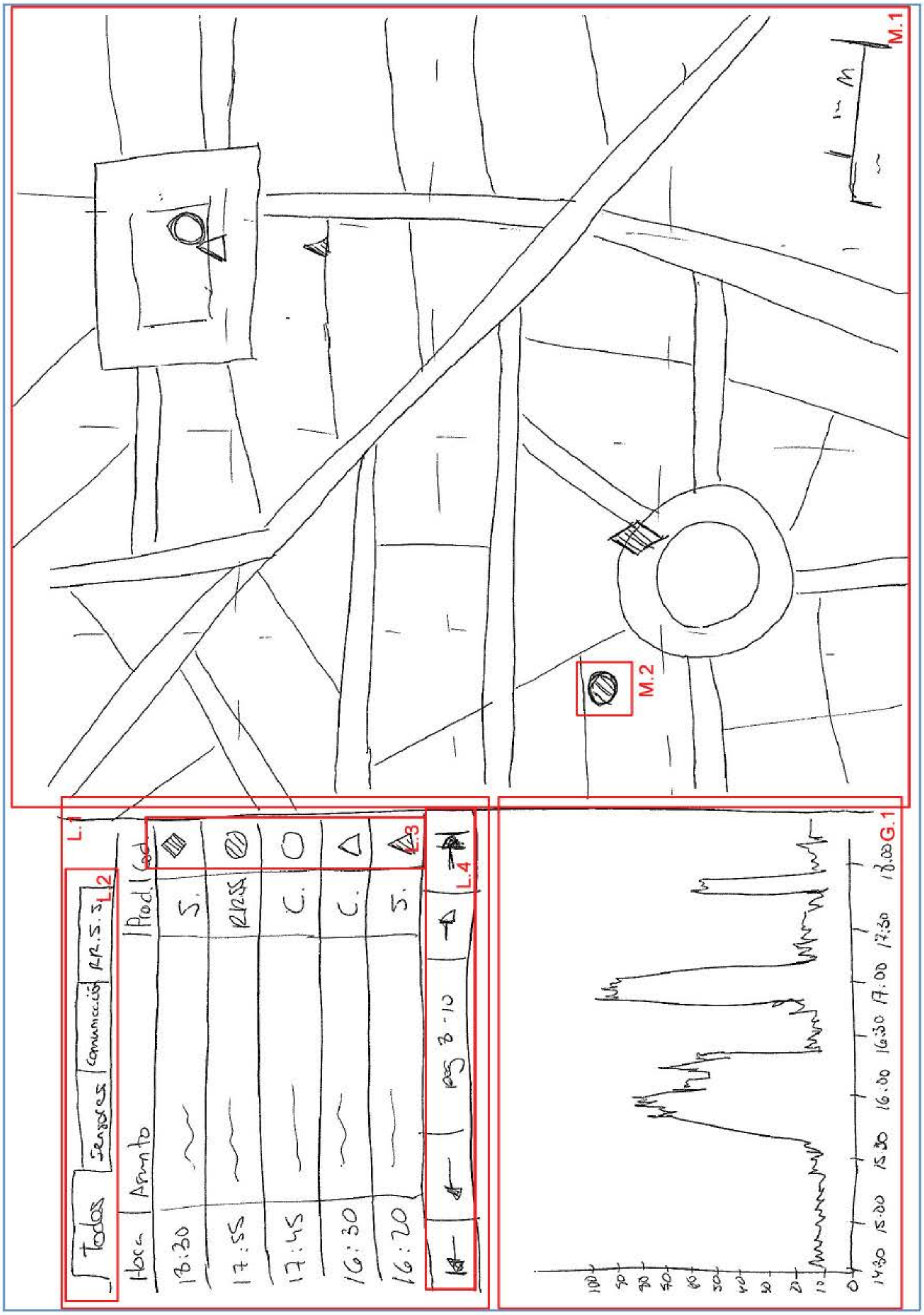
Annex Fig. C-8 Sketch created by participant 6



Annex Fig. C-9 Sketch created by participant 7



Annex Fig. C-10 Sketch created by participant 8



Annex Fig. C-11 Sketch created by participant 9

Annex D - Interactive Version of the Design Pattern Language

Annex E - List of Abbreviations

AIA	Alarm-Initiated Activities
EEMUA	Engineering Equipment and Materials Users' Association
HCi	Human-Computer Interaction
IF	Interaction Feature
IM	Interaction Mechanism
IS	Information Systems
ISA	International Society of Automation
PF	Presentation Feature
PM	Presentation Mechanism
RF	Representation Feature
RM	Representation Mechanism
SA	Situation Awareness

References (APA style)

1. ABB control systems.(n.d). In Direct Industry web page. Retrieved from http://www.directindustry.es/prod/abb-control-systems/interfaces-hombre-maquina-30259-149900.html

2. Advanced alarm and event analysis. (n.d.). In AlarmAnalyst web page. Retrieved from http://www.alarmanalyst.com/plantmonitoring.html

3. Ahmed, S. (2007). An industrial case study: Identification of competencies of design engineers. *Journal of Mechanical Design*, *129*(7), 709-716.

4. Alan, D., Janet, F., Gregory, A., & Russell, B. (2004). Human-computer interaction. *England: Pearson Education Limited*.

5. Alarm Analysis. (n.d). In GoalArt control systems web page. Retrieved from http://www.goalart.com/productsandservices/analarmanalysisexample.asp

6. Alexander, C., Ishikawa, S., & Silverstein, M. (1977). A pattern language: Towns, buildings, construction (centre for environmental structure series).

7. Alkin, M. C., & Christie, C. A. (2004). An evaluation theory tree. In M. C. Alkin (Ed.), Evaluation roots: Tracing theorists' views and influences (pp. 13-66). Thousand Oaks , CA : Sage Publications Inc.

8. Ambler, S. (1998). A realistic look at object-oriented reuse. *Software Development, 6*(1), 30-38.

9. Appleton, B. (1997). Patterns and software: Essential concepts and terminology. *Object Magazine Online, 3*(5), 20-25.

10. Bertin, J. (1983). Semiology of graphics: Diagrams, networks, maps (WJ Berg, Trans.). *Madison, WI: The University of Wisconsin Press, Ltd*.

11. Blackwell, A., & Green, T. (2003). Notational systems–the cognitive dimensions of notations framework. HCI Models, Theories, and Frameworks: Toward an Interdisciplinary Science. Morgan Kaufmann.

12. Borchers, J. O. (2001). A pattern approach to interaction design. *Ai & Society, 15*(4), 359-376.

13. Botella, F., Gallud, J. A., & Tesoreiro, R. (2011). Using interaction patterns in heuristic evaluation. In *Design, User Experience, and Usability. Theory, Methods, Tools and Practice* (pp. 23-32). Springer Berlin Heidelberg.

14. Bransby, M., Jenkinson, J., & Britain, G. (1998). *The management of alarm systems*. HSE Books.

15. Brooke, J. (1996). SUS-A quick and dirty usability scale. Usability evaluation in industry, 189(194), 4-7.

16. Bullemer, P., Reising, D. V., Burns, C., Hajdukiewicz, J., & Andrzejewski, J. (2008). In ASM (Ed.), *ASM consortium guidelines. effective operator display design.*

17. Busby, J. S., & Hibberd, R. E. (2004). Artifacts, sensemaking and catastrophic failure in railway systems. Paper presented at the *Systems, Man and Cybernetics, 2004 IEEE International Conference on, , 7* 6198-6205.

18. Card Sorting. (n.d.). In Usability.net. Retrieved from http://www.hostserver150.com/usabilit/tools/cardsorting.htm

19. Card, S. K., & Mackinlay, J. (1997). The structure of the information visualization design space. Paper presented at the *Information Visualization, 1997. Proceedings., IEEE Symposium on,* 92-99.

20. Card, S. K., Mackinlay, J. D., & Shneiderman, B. (1999). *Readings in information visualization: Using vision to think*. Morgan Kaufmann Pub.

21. Carroll, J. M., & Kellogg, W. A. (1989). Artifact as theory-nexus: Hermeneutics meets theory-based design. Paper presented at the *Conference on Human Factors in Computing Systems: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Wings for the Mind, , 1989* 7-14.

22. CAS data loggers. (n.d). In Data Logger Inc. web page. Retrieved from http://www.dataloggerinc.com/products/VR06_Paperless_Chart_Recorder/27/

23. Catapult iPower SCADA. (n.d). In Auomatech web page. Retrieved from http://www.automatech.com/products-catapult.htm

24. Chandrasegaran, S. K., Ramani, K., Sriram, R. D., Horváth, I., Bernard, A., Harik, R. F., & Gao, W. (2013). The evolution, challenges, and future of knowledge representation in product design systems. *Computer-Aided Design, 45*(2), 204-228.

25. Chen, C. (2006). *Information visualization: Beyond the horizon* Springer Science & Business.

26. Chen, H. (2004). Towards design patterns for dynamic analytical data visualization . *Proceedings of SPIE Visualization and Data Analysis.*

27. Chung, E. S., Hong, J. I., Lin, J., Prabaker, M. K., Landay, J. A., & Liu, A. L. (2004, August). Development and evaluation of emerging design patterns for ubiquitous computing. In *Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques* (pp. 233-242). ACM.

28. Cisco Prime Central system. (n.d). In Cisco systems web page. Retrieved from http://www.cisco.com/c/en/us/td/docs/net_mgmt/prime/central/1-3/user/guide/b_prime_central_13_user_guide/b_prime_central_13_user_guide_chapter_0110.html

29. ClearView-SCADA GIS. (n.d). In Relabsoft web page. Retrieved from http://www.relabsoft.com/products/clearview_-_scada_geographical_information_system_gis/

30. CYMDIST Distribution Analysis Software. (n.d). In Cooper Industries web page. Retrieved from http://www.cooperindustries.com/content/public/en/power_systems/products/power_engineeringsoftware/network_analysis/distribution_network_analysis.html

31. Dallimonti. R. (1972). Future operator consoles for improved decision-making and safety. *Instrumentation Technology, 19*(8), 23.

32. Dearden, A., & Finlay, J. (2006). Pattern languages in HCI: A critical review.*Human–computer interaction*, *21*(1), 49-102.

33. Dekker, S., & Lutzhoft, M. (2004). Correspondence, cognition and sensemaking: A radical empiricist view of situation awareness. *A Cognitive Approach to Situation Awareness: Theory and Application.Burlington, VT: Ashgate Publishing Company,* , 22-41.

34. DIASYS Netmation control system. (n.d). In Mitsubishi Hitachi Power Systems web page. Retrieved from https://www.mhps.com/en/products/detail/diasys_pro_production-control.html

35. Díaz, P., Aedo, I., Rosson, M. B., & Carroll, J. M. (2010). A visual tool for using design patterns as pattern languages. In *Proceedings of the International Conference on Advanced Visual Interfaces* (pp. 67-74). ACM.

36. Díaz, P., Rosson, M. B., Aedo, I., & Carroll, J. M. (2009). Web design patterns: Investigating user goals and browsing strategies. In *End-User Development*(pp. 186-204). Springer Berlin Heidelberg.

37. Division of Systems Analysis and Regulatory Effectiveness, Office of Nuclear Regulatory Research, US Nuclear Regulatory Commission. (2002). *Human-system interface design review guidelines*. Retrieved from http://www.bnl.gov/humanfactors/files/pdf/NUREG-0700_Rev2.pdf

38. Dynatrol Control Systems. (n.d). In Dynatrolsys web page. Retrieved from http://www.dynatrolsys.com/products/iFlex/

39. E-XD++ Entprise Suite,. (n.d). In Ucancode web page. Retrieved from http://www.ucancode.net/Products/Form2/EXDEntprise.htm

40. EEMUA - The Engineering Equipment and Materials Users' Association. (2007). In EEMUA. (Ed.), *EEMUA publication 191, alarm systems - A guide to design, management and procurement.*

41. Eick, S. G., & Karr, A. F. (2002). Visual scalability. *Journal of Computational and Graphical Statistics, 11*(1), 22-43.

42. ELSA control system. (n.d). In ELSA control systems web page. Retrieved from http://www-elsa.physik.uni-bonn.de/Beschleuniger/control_system.html

43. Emerson Process Management. (2009) . DeltaV Digital Automation System- System Overview. Retrieved November 2014 from http://www2.emersonprocess.com/siteadmincenter/PM%20DeltaV%20Documents/Brochures/DeltaV-System-Overview-v11-Brochure.pdf

44. Endsley, M. R. (2003). Designing for situation awareness: An approach to user-centered design. CRC Press.

45. Endsley, M. R., & Kris, E. O. (1994). Information presentation for expert systems in future fighters aircraft. *The International Journal of Aviation Psychology, 4*(4), 333-348.

46. Energos. (n.d). In Energos project web page. Retrieved from http://innovationenergy.org/energos/index.php

47. Flach, J., Mulder, M., & van Paassen, M. M. (2004). The concept of the situation in psychology. *A Cognitive Approach to Situation Awareness: Theory and Application,* , 42-60.

48. Fonseca, F., Egenhofer, M., Davis, C., & Câmara, G. (2002). Semantic granularity in ontology-driven geographic information systems.*Annals of Mathematics and Artificial Intelligence, 36*(1-2), 121-151.

49. Fruchter, R., & Demian, P. (2002). CoMem: Designing an interaction experience for reuse of rich contextual knowledge from a corporate memory. *Ai Edam, 16*(03), 127-147.

50. Gamble, M., & Goble, C. (2011). Quality, trust, and utility of scientific data on the web: Towards a joint model. In *Proceedings of the 3rd International Web Science Conference* (p. 15). ACM.

51. Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). *Design patterns: Elements of reusable object-oriented software.* Pearson Education.

52. Gibson, E. J. (1982). The concept of affordances in development: The renascence of functionalism. In *The concept of development: The Minnesota symposia on child psychology* (Vol. 15, pp. 55-81). Lawrence Erlbaum Hillsdale, NJ.

53. GIS Viewer. (n.d). In ETM profesional control web page. Retrieved from http://www.etm.at/index_e.asp?id=2&sb1=61&sb2=139

54. Gómez, R. R., Díez, D., Díaz, P., & Aedo, I. (2013). Situation Awareness-Oriented Alarm Visualizations: A next Step in HSC Environments. In*GRAPP/IVAPP* (pp. 483-488).

55. Guttromson, R. T., Paget, M. L., Greitzer, F. L., & Schur, A. (2007). *Human factors for situation assessment in power grid operations* Pacific Northwest National Laboratory.

56. Hafiz, M., Adamczyk, P., & Johnson, R. E. (2012). Growing a pattern language (for security). In *Proceedings of the ACM international symposium on New ideas, new paradigms, and reflections on programming and software* (pp. 139-158). ACM.

57. Heer, J., & Agrawala, M. (2006). Software design patterns for information visualization. *Visualization and Computer Graphics, IEEE Transactions on, 12*(5), 853-860.

58. Herranz, S., Romero-Gómez, R., Díaz, P., & Onorati, T. (2014). Multi-view visualizations for emergency communities of volunteers. *Journal of Visual Languages & Computing*, *25*(6), 981-994.

59. Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly, 28*(1), 75-105.

60. Hollywell, P. & Marshall, E. (1994) An Experiment to Support the Design of VDU-Based Alarm Lists for Power Plant Operators, in N. Stanton (ed.), *Human Factors in Alarm Design,* Taylor and Francis, Ltd., London, England.

61. Hughes, M. (2007). A pattern language for user assistance. Interactions , 14 (1), pp. 27- 29.

62. Information-dense and intuitive visualizations for operations, engineering and management. (n.d). In Novatech web page. Retrieved from http://www.novatechweb.com/process-control/visualization-configuration/

63. Inns, T. and Neville, P. (1998). Establishing a company-level design process to facilitate design reuse, In: Engineering Design Conference. Brunel University.

64. International Society of Automation (ISA). (2009). Standard No. ANSI/ISA-18.2-2009 Management of Alarm Systems for the Process Industries.

65. IrisView control system. (n.d). In Tekcomms web page. Retrieved from http://www.tekcomms.com/irisview-network-visibility

66. ISO/IEC TR 9126-4:2004. (n.d). Retrieved from lhttp://www.iso.org/iso/catalogue_detail.htm?csnumber=39752

67. Jalote-Parmar, A., & Badke-Schaub, P. (2010). Situation awareness in medical visualization to support surgical decision making. Paper presented at the *Proceedings of the 28th Annual European Conference on Cognitive Ergonomics,* 201-208.

68. Kerren, A., Ebert, A., & Meyer, J. (2007). *Human-centered visualization environments* Springer-Verlag.

69. Kitchenham, B., Pearl Brereton, O., Budgen, D., Turner, M., Bailey, J., & Linkman, S. (2009). Systematic literature reviews in software engineering–a systematic literature review. *Information and Software Technology, 51*(1), 7-15.

70. Klein, G., Moon, B., & Hoffman, R. R. (2006). Making sense of sensemaking 1: Alternative perspectives. *Intelligent Systems, IEEE, 21*(4), 70-73.

71. Klein, G., Moon, B., & Hoffman, R. R. (2006). Making sense of sensemaking 2: A macrocognitive model. *Intelligent Systems, IEEE, 21*(5), 88-92.

72. Köhler, W. (1970). Gestalt psychology: An introduction to new concepts in modern psychology. WW Norton & Company.

73. Kolfschoten, G. L., & Veen, W. (2005, January). Tool support for GSS session design. In *System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on* (pp. 16b-16b). IEEE.

74. Kolfschoten, G., Lukosch, S., Verbraeck, A., Valentin, E., & de Vreede, G. J. (2010). Cognitive learning efficiency through the use of design patterns in teaching. *Computers & Education*, *54*(3), 652-660.

75. Kortland, D., & Kragt, H. (1980). Process alerting systems as a monitoring tool for the operation. *Loss Prevention and Safety Promotion in the Process Industry*, *3*, 804-814.

76. Krueger, C. W. (1992). Software reuse. *ACM Computing Surveys (CSUR)*, *24*(2), 131-183.

77. Kubo, A., Washizaki, H., & Fukazawa, Y. (2007). A metric for measuring the abstraction level of design patterns. In *Proceedings of the 14th Conference on Pattern Languages of Programs* (p. 19). ACM.

78. Landay, J. A., & Borriello, G. (2003). Design patterns for ubiquitous computing. *Computer, 36*(8), 93-95.

79. Lees, F. P. (1974). Research on the process operator. *The Human Operator in Process Control,* 386-425.

80. Lipshitz, R. (1993) Converging themes in the study of decision making in realistic settings in G. A. Klein, J. Orasanu, R. Calderwood and C. E. Zsambok (Eds.) Decision Making in Action: Models and Methods (pp. 103-137), New Jersey: Ablex

81. Lohs, S., Karnapke, R., Nolte, J., & Lagemann, A. (2012). Self-stabilizing sensor networks for emergency management. Paper presented at the *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on,* 715-720.

82. Lorincz, K., Malan, D. J., Fulford-Jones, T. R., Nawoj, A., Clavel, A., Shnayder, V., Shnayder, V., Mainland, G., Welsh, M., & Moulton, S. (2004). Sensor networks for emergency response: Challenges and opportunities. *Pervasive Computing, IEEE, 3*(4), 16-23.

83. Mackinlay, J. (1986). Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics (TOG), 5*(2), 110-141.

84. Markus, M. L. (2001). Toward a theory of knowledge reuse: Types of knowledge reuse situations and factors in reuse success. *Journal of Management Information Systems, 18*(1), 57-94.

85. Mattiasson, C. (1999). The alarm system from the operator's perspective. Paper presented at the *Human Interfaces in Control Rooms, Cockpits and Command Centres, 1999. International Conference on,* 217-221.

86. McInerney, P. (2002): UI Patterns. Patterns Workshop, CHI2002. 13.1-13.5

87. Mikkelsen, C., Johansson, J., & Rissanen, M. (2011). Interactive information visualization for sensemaking in power grid supervisory systems. Paper presented at the *Information Visualisation (IV), 2011 15th International Conference on,* 119-126.

88. Montero Moreno, S. (2005). Integración de patrones en el proceso de diseño de sistemas hipermedia mediante el uso de ontologías (Doctoral dissertation). Retrieved from Universidad Carlos III de Madrid e-archivo (URI: http://hdl.handle.net/10016/671)

89. Moon Suh, S., Sook Jang, G., Ok Park, G., Yun Park, H., & Soo Koo, I. (2006). An elastic display method for visualizing and navigating a large quantity of alarms in a control room of a nuclear power plant. Advances in Systems, Computing Sciences and Software Engineering, 1-3.

90. Munzner, T. (2002). Information visualization. *IEEE Computer Graphics and Applications, 22*(1), 0020-21.

91. Munzner, T. (2009). A nested model for visualization design and validation. *Visualization and Computer Graphics, IEEE Transactions on, 15*(6), 921-928.

92. Nielsen, J. (1994, April). Usability inspection methods. In *Conference companion on Human factors in computing systems* (pp. 413-414). ACM.

93. Niwa, Y., & Hollnagel, E. (2001). Enhancing operator control by adaptive alarm presentation. *International Journal of Cognitive Ergonomics*, *5*(3), 367-384.

94. Norman, D. A. (1993). Things that make us smart: Defending human attributes in the age of the machine. Basic Books.

95. Oak Ridge National Laboratory (2010). Best Ever Alarm System Toolkit. Retrieved from http://www.aps.anl.gov/epics/meetings/2010-10/14/BEAST_EpicsMeeting2010.pdf

96. Offermann, P., Levina, O., Schönherr, M., & Bub, U. (2009). Outline of a design science research process. Paper presented at the *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology,* 7.

97. Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems, 24*(3), 45-77.

98. PowerOn Reliance Energy Management.(n.d). In General Electric Digital Energy web page. Retrieved from https://www.gedigitalenergy.com/uos/ems.htm#ems2

99. Precision Digital Corporation Systems. (n.d). In Predig web page. Retrieved from http://www.predig.com/products/annunciators.php

100. Prestopnik, N. (2013). Design Science in Human-Computer Interaction: A Model and Three Examples. iSchool Information Science and Technology-Dissertations. Paper 83.

101. Pries-Heje, J., Baskerville, R., & Venable, J. R. (2008). Strategies for design science research evaluation. ECIS(2008).

102. Proface control systems. (n.d). In Proface web page. Retrieved from http://www.proface.es/product/smart_portal/sp5000/features/features_4.html

103. Ralph, P. (2010). Fundamentals of software design science. Sauder School of Business: 177. Vancouver, Canada: University of British Columbia.

104. Rogers, P. J., & Fraser, D. (2003). Appreciating appreciative inquiry. *New Directions for Evaluation*, *2003*(100), 75-83.

105. Romero-Gómez, R., Diez, D., Díaz, P., & Aedo, I. (2013). Situation awareness-oriented alarm visualizations: A next step in HSC environments. *4th International Conference of Information Visualization Theory and Applications,* Barcelona.

106. Romero-Gómez, R., Tena, S., Díez, D., & Díaz, P. (2013). The Application of Situation Awareness-Oriented Design to the Smart Grid Domain. Actas del XIV Congreso Internacional de Interacción Persona Ordenador.

107. Rothenberg, D. H. (2009). Alarm management for process control: A best-practice guide for design, implementation, and use of industrial alarm systems McGraw-Hill Education.

108. Rouse, W. (1983). Models of human problem solving: Detection, diagnosis, and compensation for system failures. *Automatica, 19*(6), 613-625.

109. Rousseau, R., Tremblay, S., & Breton, R. (2004). Defining and modeling situation awareness: A critical review. *A Cognitive Approach to Situation Awareness: Theory and Application,* , 3-21.

110. Salingaros, N. A. (2000). The structure of pattern languages. *Architectural Research Quarterly, 4*(02), 149-162.

111. Salustri, F. A. (2005). Using pattern languages in design engineering. In DS 35: Proceedings ICED 05, the 15th International Conference on Engineering Design, Melbourne, Australia, 15.-18.08. 2005.

112. Sandblad, B., Andersson, A. W., Byström, J., & Kauppi, A. (2002). New control strategies and user interfaces for train traffic control. *Computers in Railways VIII.*

113. SCADA software (n.d). In SCADA software web page. Retrieved from http://www.scadasoftware.net/scadasoftware/

114. Scriven, M. S. (1967). The methodology of evaluation (Vol. 1). Chicago : Rand McNally.

115. Sedig, K. & Parsons, P. (2013). Interaction design for complex cognitive activities with visual representations: A pattern-based approach. AIS Transactions on Human-Computer Interaction, 5(2), 84–133.

116. Seemann, N. (2014). *A Design Pattern Oriented Programming Environment* (Document dissertation). Retrieved from diplom. De (URI: http://www.diplom.de/e-book/217760/a-design-pattern-oriented-programming-environment)

117. Segerståhl, K., & Jokela, T. (2006). Usability of interaction patterns. In *CHI'06 extended abstracts on Human factors in computing systems* (pp. 1301-1306). ACM.

118. Sharmin, M., Bailey, B. P., Coats, C., & Hamilton, K. (2009). Understanding knowledge management practices for early design activity and its implications for reuse. Paper presented at the *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems,* 2367-2376.

119. Sheridan, T. B. (1992). Telerobotics, automation, and human supervisory control. MIT press.

120. Shneiderman, B. (1996). The eyes have it: A task by data type taxonomy for information visualizations. Paper presented at the *Visual Languages, 1996. Proceedings., IEEE Symposium on,* 336-343.

121. Siemens AG. (2009). *The SIMATIC PCS 7 Process Control System.* Retrieved from http://www.intea.hr/downloads/simatic_pcs_7.pdf

122. Singh, S. K. (2003). *Industrial Instrumentation & Control, 2e*. Tata McGraw- Hill Education.

123. Smart Grid Monitoring-Geographic Information Systems (GIS). (n.d.). In ETAP web page. Retrieved from http://etap.com/smart-grid/smart-grid-dispatching-gis.htm

124. Spence, R. (2007). *Information visualization: Design for interaction* (2nd Edition ed.). Pearson-Prentice Hall.

125. Stanton, N. A. (1994). Alarm initiated activities. *Human Factors in Alarm Design,* , 93-117.

126. Stanton, N. A., & Stammers, R. B. (1998). Alarm initiated activities: matching formats to tasks.

127. Stolte, C., Tang, D., & Hanrahan, P. (2003). Multiscale visualization using data cubes. *Visualization and Computer Graphics, IEEE Transactions on, 9*(2), 176-187.

128. Sutcliffe, A. G. (2000). On the Effective Use and Reuse of HCI Knowledge. ACM Transactions on Computer-Human Interaction (TOCHI) , 7 (2), 197-221.

129. Sutcliffe, A. G., & Carroll, J. M. (1999). Designing Claims for Reuse in Interactive Systems Design. International Journal of Human-Computer Studies , 50 (3), 213-241.

130. Syncaps V3. (n.d.). In Syntagm web page. Retrieved from https://www.syntagm.co.uk/design/cardsortdl.shtml#syncapsv3

131. The Gestalt principles. (n.d.). In Spokane Falls Community College web page about graphic design. Retrieved from http://graphicdesign.spokanefalls.edu/tutorials/process/gestaltprinciples/gestaltprinc.htm

132. Thomas, J. J., & Cook, K. A. (2005). Illuminating the path: The research and development agenda for visual analytics IEEE Computer Society Press.

133. Todd, E., Kemp, E., & Phillips, C. (2004). What makes a good user interface pattern language? Paper presented at the *Proceedings of the Fifth Conference on Australasian User Interface-Volume 28,* 91-100.

134. Toshiba SCADA system. (n.d). In Toshiba web page. Retrieved from http://www.toshiba.co.jp/sis/en/tands/control/tos_con.htm

135. Tversky, B., Morrison, J. B., & Betrancourt, M. (2002). Animation: Can it facilitate? *International Journal of Human-Computer Studies, 57*(4), 247-262.

136. Van Duyne, D. K., Landay, J. A., & Hong, J. I. (2003). The design of sites: Patterns, principles, and processes for crafting a customer-centered web experience Addison-Wesley Professional.

137. Van Welie, M., & Van der Veer, G. C. (2003). Pattern languages in interaction design: Structure and organization. In *Proceedings of interact* (Vol. 3, pp. 1-5).

138. Venable, J., Pries-Heje, J., & Baskerville, R. (2012). A comprehensive framework for evaluation in design science research. *Design science research in information systems. advances in theory and practice* (pp. 423-438) Springer.

139. Vis Javascript library. (n.d). Retrieved from http://visjs.org/

140. Visser, W. (1995). Reuse of knowledge: Empirical studies. *Case-based reasoning research and development* (pp. 335-346) Springer.

141. Ware, C. (2012). *Information visualization: Perception for design*. Elsevier.

142. Wassink, I., Kulyk, O., van Dijk, B., van der Veer, G., & van der Vet, P. (2009). Applying a user-centered approach to interactive visualisation design. *Trends in interactive visualization* (pp. 175-199) Springer.

143. Westinghouse Electric Company LLC. (2011). *Westinghouse Alarm Presentation System.* Retrieved from http://westinghousenuclear.com/Portals/0/Operating%20Plant%20Services/Automation/Human-System%20Interface/NA-0057%20Alarm%20Presentation%20System.pdf

144. Wilkinson, L., Wills, D., Rope, D., Norton, A., & Dubbs, R. (2006). *The grammar of graphics*. Springer Science & Business Media.

145. Xie, Z., Guo, Z., Ward, M. O., & Rundensteiner, E. A. (2010). Operator-centric design patterns for information visualization software. Paper presented at the *IS&T/SPIE Electronic Imaging,* 75300J-75300J-12.

146. Y-Plant Alert Alarm Annunciator System. (n.d). In Yokogawa web page. Retrieved from http://www.yokogawa.com/us/products/systems-integration/yokogawa-integrated-solutions/y-plant-alert.htm

147. Zhang, J., Johnson, K. A., Malin, J. T., & Smith, J. W. (2002). Human-centered information visualization. Paper presented at the *International Workshop on Dynamic Visualizations and Learning, Tubingen, Germany.*

148. Zhu, J., Zhuang, E., Ivanov, C., & Yao, Z. (2011). A data-driven approach to interactive visualization of power systems. *Power Systems, IEEE Transactions on*, *26*(4), 2539-2546.

149. Zwaga, H., & Hoonhout, H. (1995). Supervisory control behaviour and the implementation of alarms in process control. Paper presented at the Human Factors in Alarm Design, 119-134.

Concerning to the usage of elements of the design pattern language, Fig. 5.2-18 shows the mean of participants' responses to the closed-ended questions. It displays that participants found the design pattern language as useful to create the proposed solution to the given design scenario. In particular, the highest agreement (coloured in green) with the question at hand, was obtained by Q7 *("The design pattern language guided me to design the proposed solution")* and the lowest scores, which reflects a strongly disagreement with the question at hand, were obtained by both Q1 *("Given the design scenario, the selection of design patterns was based on my intuition")* and Q8 *("After selecting a design pattern, looking for new design solutions was based on my intuition").* These results suggest that the design pattern language addresses the lack of experience of non-experienced designers in alarm visualization design by both providing them design solutions to design problems they have been requested to face and guiding them to connected design ideas to the problems at hand.

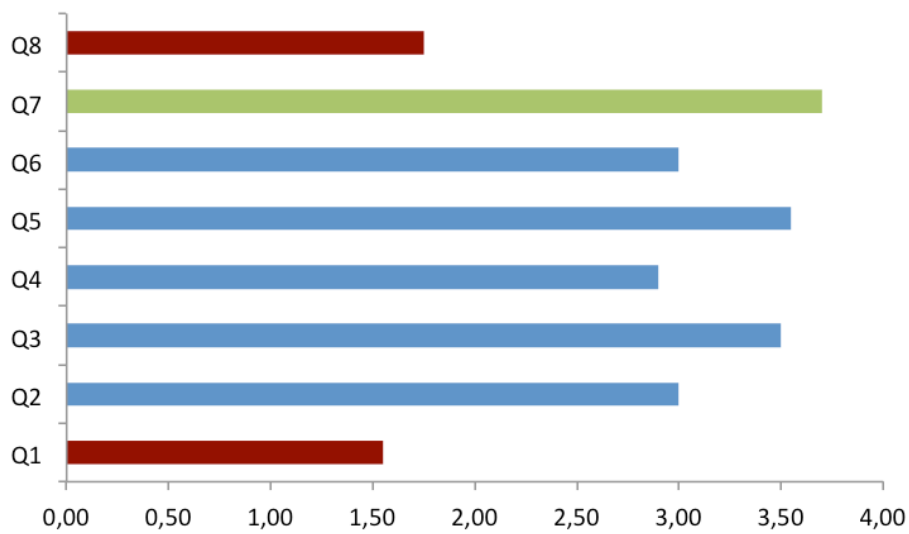


Fig. 5.2-18 Second round of the experimental evaluation: Results of the efficacy questions on a scale 1-4 (4=strongly agree); for questions use Table 5.2-14

To argument these promising conclusions about the usability of the design pattern language, the usage of design patterns in the sketches and the participants' responses to the open-ended questions displayed in Table 5.2-15 were analysed. Concerning to the usage of design patterns in the sketches, the analysis of the results followed the same strategy than in the first round of the evaluation. Although these results displayed in Fig. 5.2-19 are quite similar to those obtained in the first round of the evaluation (see Fig.

5.2-13), two main differences should be underlined. The first one is related to the absence of design patterns applied incorrectly. The second one is related to the increasing application of all strongly recommended design patterns such as the case of *IM(8) Paging*.

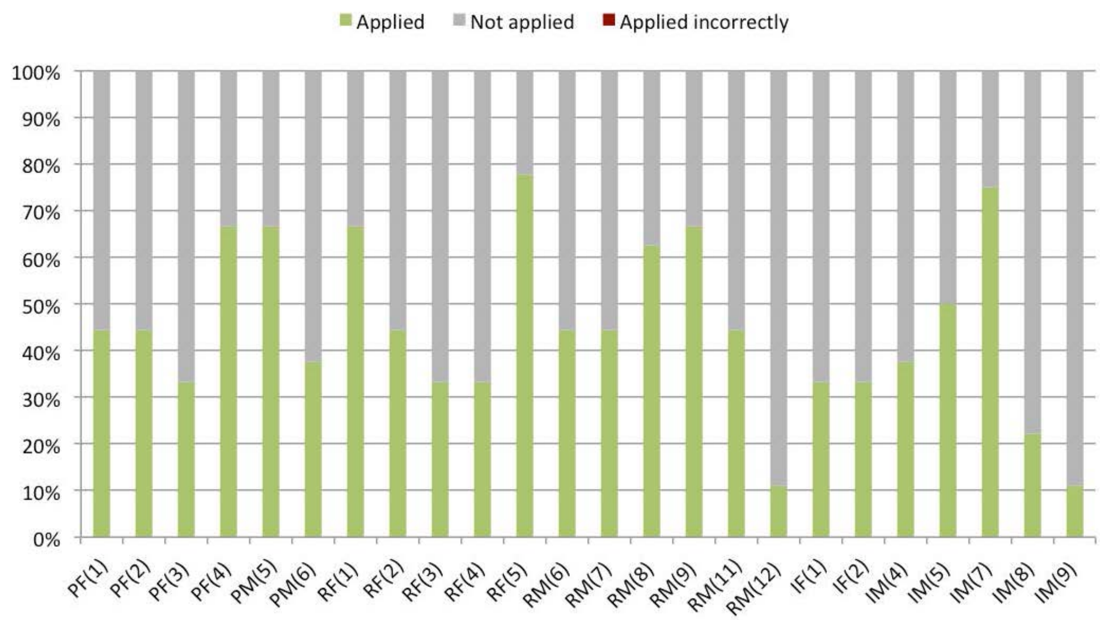


Fig. 5.2-19 Second round of the Experimental evaluation: Results on the design patterns used; for patterns' names use Table 5.2-12

The analysis of the results obtained from the participants' responses to the open-ended questions helped to clarify these two main differences. To these questions, most of participants not only reported about the utility of some elements of design patterns such as the design problems descriptions, the visual examples, and the relations between design patterns to create a design solution but also about the capabilities provided by the interactive version of the pattern language. In particular, all participants appreciated the interactive capabilities provided by such interactive version such as *filtering* and *hovering* on a specific design pattern in order to navigate more easily through the design pattern language. They found particularly useful the capability of quickly consulting visual examples of the proposed solution by hovering on each design pattern. In this way, these results seem to suggest that, in contrast to the previous experience of participants using a paper version of the pattern language, in this second round of the evaluation participants were able to spend more time understanding the descriptions of both the design problems and solutions proposed by design patterns instead of having to understand the structure

PRESENTATION PATTERNS

PF(1) DETAILS HIERARCHY

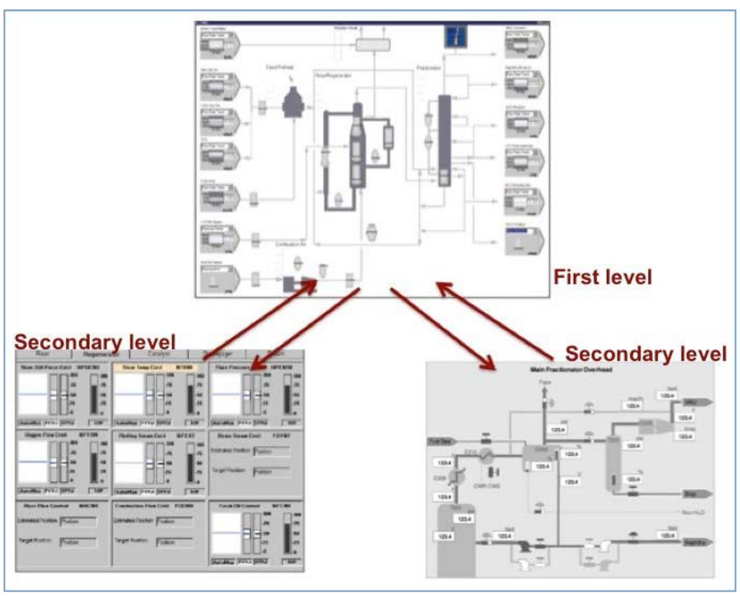
Classification. Purpose: Presentation. Level of abstraction: Feature.

Context. Within his area of responsibility and authority, the human operator needs to comprehend sets of incoming alarms displayed by alarm displays. Based on these displays, he would like to explore the overall alarm information space according to different alarm dimensions such as typology, priority, or location.

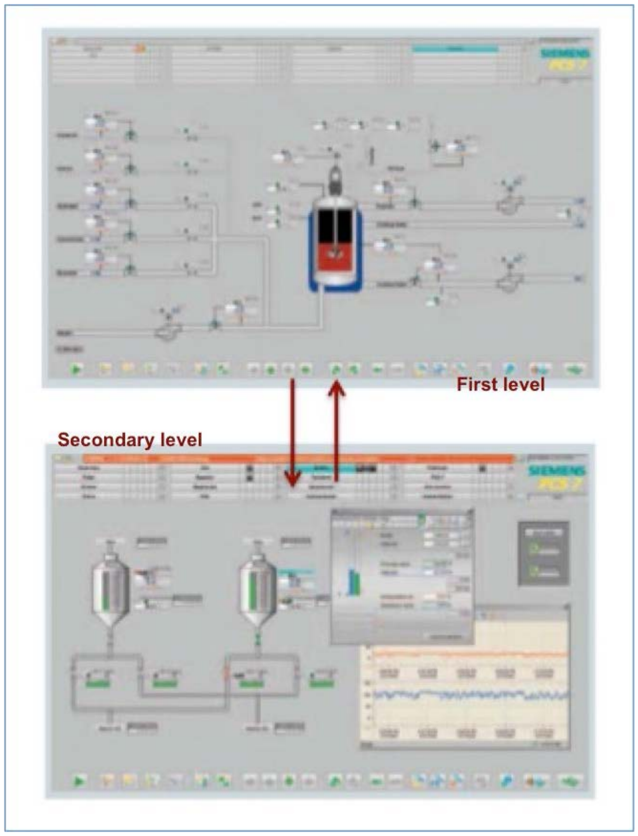
Problem. *The human operator needs to explore the overall alarm information space according to different alarm dimensions in order to diagnose the cause of a failure in the controlled process.*

Solution. Use a standard display hierarchy to present the multi-level views necessary for exploring alarm information. These levels follow an expected progression from the general to the more detailed in order to aid the operator in performing different tasks. A single first level can be associated with several secondary levels. Each secondary level can only be associated therefore with a single first level.

Known uses.



Annex Fig. A-1 An implementation of the Details hierarchy pattern: Hierarchy of levels applied to a control system interface for a power plant [107]



Annex Fig. A-2 An implementation of the Details Hierarchy pattern: Hierarchy of levels applied to the SIMATIC PCS 7 process control system developed by Siemens [121]

Rationale. A display hierarchy delivers a robust structure that encourages ready access to information while at the same time keeping important situation context and promoting efficient navigation to go deeper [107].

Relations.

- Composition relationship: *PF(3) Primary level of detail, PF(4) Secondary level of detail.*
- Specialization relationship: *PM(5) Overview and detail*

PF(2) Spatial dedication and continous visibility

Classification. Purpose: Presentation. Level of abstraction: Feature.

Context. Under all controlled process conditions, even during minor disturbances, the human operator would like to know all key alarms that are directly safety related and important process alarms related to safety critical systems.

Concerning to the usage of elements of the design pattern language, Fig. 5.2-18 shows the mean of participants' responses to the closed-ended questions. It displays that participants found the design pattern language as useful to create the proposed solution to the given design scenario. In particular, the highest agreement (coloured in green) with the question at hand, was obtained by Q7 *("The design pattern language guided me to design the proposed solution")* and the lowest scores, which reflects a strongly disagreement with the question at hand, were obtained by both Q1 *("Given the design scenario, the selection of design patterns was based on my intuition")* and Q8 *("After selecting a design pattern, looking for new design solutions was based on my intuition").* These results suggest that the design pattern language addresses the lack of experience of non-experienced designers in alarm visualization design by both providing them design solutions to design problems they have been requested to face and guiding them to connected design ideas to the problems at hand.

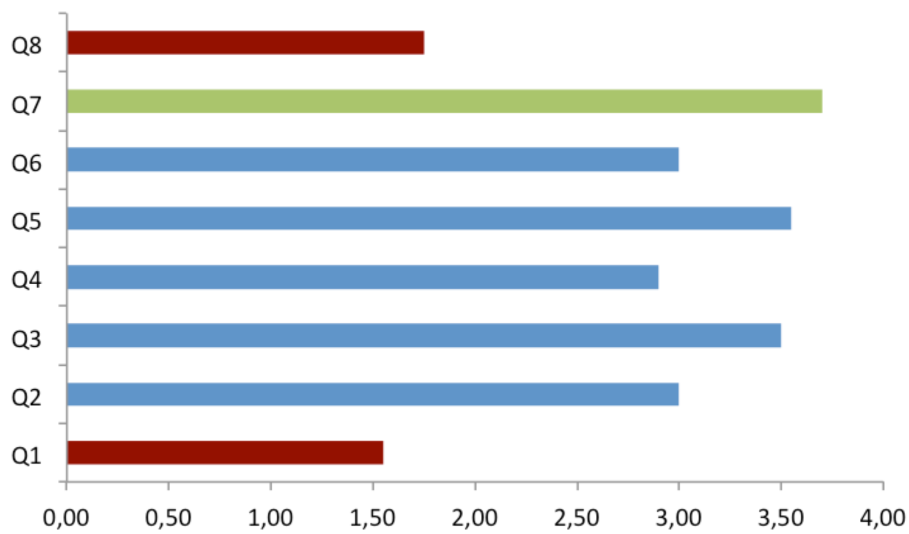


Fig. 5.2-18 Second round of the experimental evaluation: Results of the efficacy questions on a scale 1-4 (4=strongly agree); for questions use Table 5.2-14

To argument these promising conclusions about the usability of the design pattern language, the usage of design patterns in the sketches and the participants' responses to the open-ended questions displayed in Table 5.2-15 were analysed. Concerning to the usage of design patterns in the sketches, the analysis of the results followed the same strategy than in the first round of the evaluation. Although these results displayed in Fig. 5.2-19 are quite similar to those obtained in the first round of the evaluation (see Fig.

5.2-13), two main differences should be underlined. The first one is related to the absence of design patterns applied incorrectly. The second one is related to the increasing application of all strongly recommended design patterns such as the case of *IM(8) Paging*.

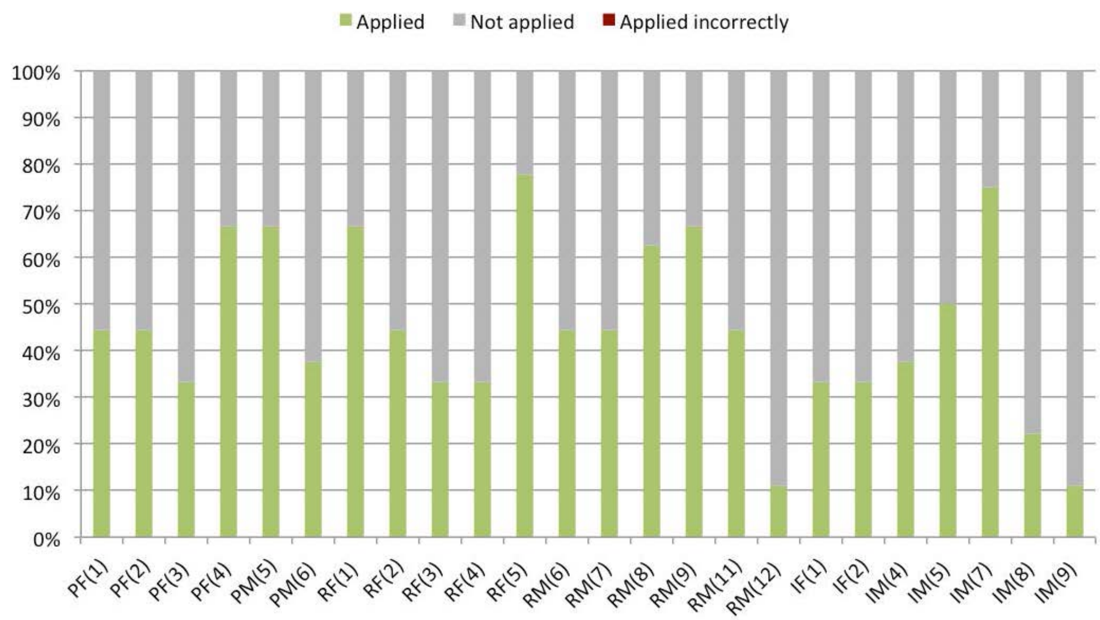


Fig. 5.2-19 Second round of the Experimental evaluation: Results on the design patterns used; for patterns' names use Table 5.2-12

The analysis of the results obtained from the participants' responses to the open-ended questions helped to clarify these two main differences. To these questions, most of participants not only reported about the utility of some elements of design patterns such as the design problems descriptions, the visual examples, and the relations between design patterns to create a design solution but also about the capabilities provided by the interactive version of the pattern language. In particular, all participants appreciated the interactive capabilities provided by such interactive version such as *filtering* and *hovering* on a specific design pattern in order to navigate more easily through the design pattern language. They found particularly useful the capability of quickly consulting visual examples of the proposed solution by hovering on each design pattern. In this way, these results seem to suggest that, in contrast to the previous experience of participants using a paper version of the pattern language, in this second round of the evaluation participants were able to spend more time understanding the descriptions of both the design problems and solutions proposed by design patterns instead of having to understand the structure

PRESENTATION PATTERNS

PF(1) DETAILS HIERARCHY

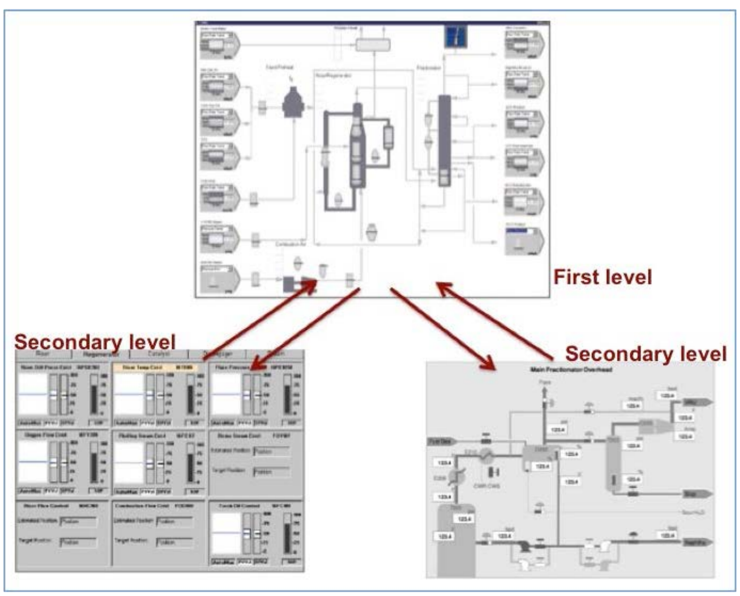
Classification. Purpose: Presentation. Level of abstraction: Feature.

Context. Within his area of responsibility and authority, the human operator needs to comprehend sets of incoming alarms displayed by alarm displays. Based on these displays, he would like to explore the overall alarm information space according to different alarm dimensions such as typology, priority, or location.

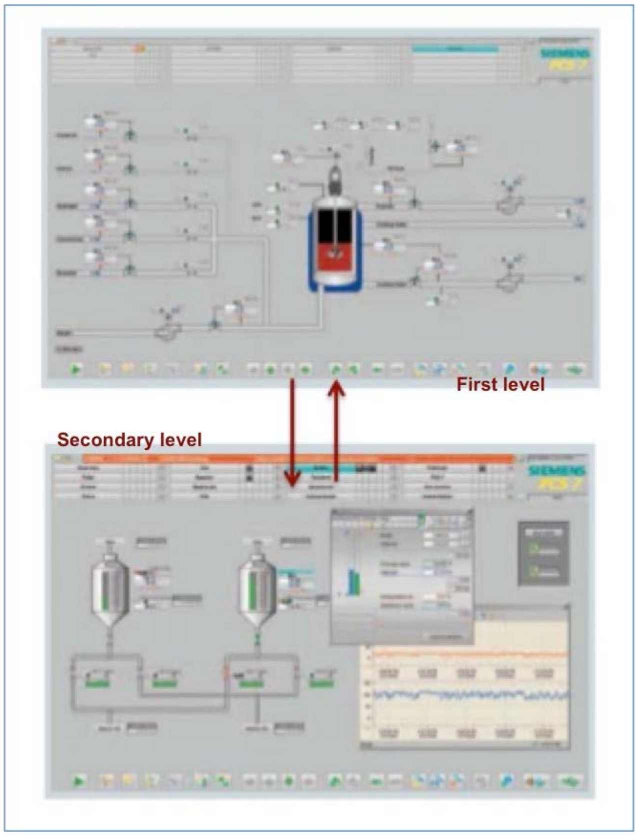
Problem. *The human operator needs to explore the overall alarm information space according to different alarm dimensions in order to diagnose the cause of a failure in the controlled process.*

Solution. Use a standard display hierarchy to present the multi-level views necessary for exploring alarm information. These levels follow an expected progression from the general to the more detailed in order to aid the operator in performing different tasks. A single first level can be associated with several secondary levels. Each secondary level can only be associated therefore with a single first level.

Known uses.



Annex Fig. A-1 An implementation of the Details hierarchy pattern: Hierarchy of levels applied to a control system interface for a power plant [107]



Annex Fig. A-2 An implementation of the Details Hierarchy pattern: Hierarchy of levels applied to the SIMATIC PCS 7 process control system developed by Siemens [121]

Rationale. A display hierarchy delivers a robust structure that encourages ready access to information while at the same time keeping important situation context and promoting efficient navigation to go deeper [107].

Relations.

- Composition relationship: *PF(3) Primary level of detail, PF(4) Secondary level of detail.*
- Specialization relationship: *PM(5) Overview and detail*

PF(2) Spatial dedication and continous visibility

Classification. Purpose: Presentation. Level of abstraction: Feature.

Context. Under all controlled process conditions, even during minor disturbances, the human operator would like to know all key alarms that are directly safety related and important process alarms related to safety critical systems.