Architectural patterns and quality attributes interaction

Gianantonio Me*†, Coral Calero *, Patricia Lago†
{gianantonio.me, coral.calero}@uclm.es; {g.me,p.lago}@vu.nl
*Escuela Superior de Informatica, Universidad de Castilla La Mancha, Ciudad Real, Spain
†Department of Computer Science, Vrije Universiteit Amsterdam, The Netherlands

Abstract— **Architectural patterns and styles represent common solutions to recurrent problems. They encompass architectural knowledge about how to achieve holistic system quality. The relation between patterns (or styles) and quality attributes has been regularly addressed in the literature. However, there is a lack of a consolidated and systematically built reference capturing this relation and eventually making it available for reuse in the form of patterns-quality attributes knowledge. If captured, this knowledge can be used as an architectural decision framework where solutions (patterns) are strongly intertwined with quality (attributes). Such a framework should also contemplate variants and combinations of patterns. In order to create the framework, we first proposed a *lightweight theory* on the interaction of patterns/styles and quality attributes. That framework has been built by starting from a key study on the interaction between architectural patterns and quality attributes. We then challenged the theory against the systematic survey of a pilot set of primary studies. This paper presents the preliminary results of this survey on architectural patterns and styles, and their interaction with quality attributes. The preliminary results show that the initial theory can work as a platform for integrating the body of knowledge gathered through the analysis.**

Keywords— Architectural patterns, architectural styles, quality attributes.

I. INTRODUCTION

Architectural patterns and styles are recurrent solutions to common problems. Among others, they include knowledge on quality attributes [1]. For the sake of simplicity, throughout the paper we use the term "architectural pattern". In fact, according to Buschmann [2], patterns and styles are very similar as every architectural style can be described as an architectural pattern. However, some differences can be considered as essential, the most relevant being that patterns are more problem oriented, while styles do not refer to a specific design situation. Accordingly, in our analysis we tried to make explicit if and why authors adopt the term "pattern" or "style".

We observe a similar problem with the definition of quality attribute. Again, for the sake of simplicity, we adopt the term "quality attribute". In our analysis, if necessary, we make explicit the term used by the authors such as non-functional requirement, quality property, quality dimension, etc.

In the literature, patterns have been usually described according to the functionality they deliver and the strength or liability showed with respect to several quality attributes. According to [1] strengths or liabilities assess the importance of the impact of patterns on quality attributes. For instance: a key strength or key liability determines if to use or to avoid a pattern in a specific situation. In this line of reasoning, the degree which patterns impact on quality might determine architectural choices (i.e. adopting or avoiding a pattern for a given design problem). Still, many challenges are still open. For instance, variants of patterns and their interaction with quality attributes are mostly unexplored. Similarly, we know very little about how combinations of patterns jointly influence the overall quality of a system. Moreover, the "interaction" between patterns and quality attributes has been described in the literature as impact or influence but the associated semantics is still theoretically undefined. In this work, we are interested in understanding if and how patterns impact on quality attributes, how and to what extent it is possible to measure such impact, and if found interactions show unknown properties.

Architectural patterns include knowledge on quality attributes. Architects rely on that knowledge for effective architectural decision-making. Increasing that knowledge means increasing the role of patterns in satisfying quality attributes. This study has its starting point in a preliminary literature-based *lightweight theory (Fig. 1)*, followed by a systematic literature review (SLR) of a subset of primary studies. The theory is challenged against the results of the SLR and incrementally revisited. The aim of this paper is to present our preliminary results of the SLR, hence providing a first glimpse on how patterns and quality attributes interact and starting the discussion in the software architecture community.

This paper is organized as follows: Section II presents the study design. Section III depicts the analysis of a randomly selected sub-set of primary studies. Section IV summarizes how the SLR findings challenge our theory, while Section V presents conclusions and future work.

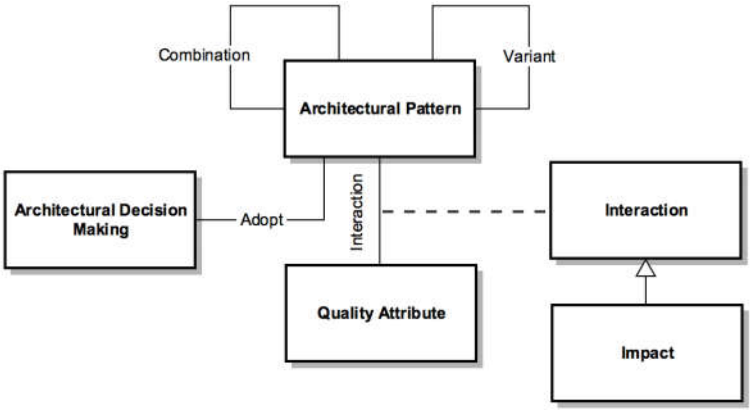

Fig. 1-Lightweight Theory

II. STUDY DESIGN

This section presents the essential aspects of our study design, the *lightweight theory* and the SLR.

A. Study Design

This study uses a two steps approach by building a theory and then lead a SLR. Firstly, a *lightweight theory* has been created according to [1]. Secondly, a SLR protocol has been developed. Results from primary studies are analyzed in order to challenge the theory. This study reports on a preliminary analysis carried out on a sub-set of primary studies.

B. A lightweight theory on Architectural Patterns and Quality Attributes

Fig.1 illustrates the basic concepts part of our theory (by means of a UML class diagram). The model reflects the knowledge reported in [1]. In the following we describe the extracted model elements. To have a clear and reusable starting point, we adopted an initial body of knowledge, with explicit boundaries, given by [1] (from here on, referred to as the "main literature source" or "main study"). We decided to start from this main study because:

1) It explicitly addresses the interaction between architectural patterns and quality attributes.

2) It has its roots in the classic work of Buschmann [2], which in turn is used as major reference in the field.

3) Has a reference quality model (Standard ISO/IEC 9126 [3]).

Our initial theory does not consider known studies outside the main source, except for the knowledge quoted by the authors. On the one hand, it is unrealistic to aim at a "universally agreed-upon" body of knowledge. In particular, in the main source most of the knowledge on patterns reported in [2] has been widely taken in account. Also, by adopting the Standard ISO/IEC 9126 as quality model, the authors took into account a second source of knowledge already.

Therefore, this main study can be clearly considered as a very stable and representative starting point for our theory.

By coding the main study we partitioned the knowledge gathered in conceptual areas. As illustrated in Fig. 1, each class represents a conceptual area. Namely, architectural decision-making results in the adoption of architectural patterns – see association Adopt. Architectural Patterns can be in a "pure form", or further specialized into variants or composed into hybrids – see associations Variant or Combination, respectively. Quality attributes do interact with architectural patterns – see association class Interaction. According to the main study, this interaction can be further considered as an Impact (on selected quality attributes, and due to the adoption of that pattern). By coding the main study we have captured the following knowledge into the main conceptual areas:

Architectural Decision Making. In a social perspective, this area pertains to a broader conceptual area referred to as "architectural practices". This aims at leveraging the whole experience of architects. For the goal of this work the focus has been narrowed to the decision-making area. The rationale of this choice is that we are interested to capture how increasing the knowledge on patterns and quality attributes does support better decisions. Therefore, decision-making assumes a key role in architecting practice. The authors of the main study consider architectural patterns and quality in a practitioner's scenario. The overall perspective of the main study is grounded in decision-making and architectural knowledge management. The main focus is on decisions and how they can help achieving target values of the quality attributes. The model depicted in Fig. 1, hence, generalizes from many aspects about Architectural Decision Making, like for instance, from the fact that architectural decisions may have a specification, may lead to failures, and may overlook consequences that possibly impact on the quality of the whole system. There is a strong emphasis on failure scenarios and the analysis of emerging liabilities. By increasing the knowledge on patterns it is possible to make better-informed decisions, avoid failures and better satisfy quality attributes and achieve system wide quality targets.

TABLE I. ARCHITECTURAL PATTERNS AND QUALITY ATTRIBUTES

Architectural Patterns	Quality Attributes[a]						
	Usability	*Security*	*Maintainability*	*Efficiency*	*Reliability*	*Portability*	*Implementability*
Layered	=	++	++	-	+	+	-
Pipes & Filters	-	-	+	+	- -	++	-
Blackboard	=	-	++	-	=	=	--
Model View Controller	++	=	-	-	=	-	-
Presentation Abstraction Control	+	=	+	--	=	+	--
Microkernel	=	=	++	--	+	++	--
Reflection	=	=	++	-	--	+	-
Broker	+	+	+	=	=	++	+

a. + Stands for "Strength", ++ for "Key strength". – for "Liability", -.- stands for "Key liability" according to [1]

In this line of reasoning the usefulness of architectural patterns might be improved by capturing the architectural knowledge on the interaction between patterns and quality attributes. Architectural patterns are considered as "vehicles" for capturing architecture rationale.

Architectural Pattern. As seen before, patterns have been identified following the classic work of Buschmann [2]. Accordingly, patterns can be characterized by their impact (positive or negative) on specific quality attributes. This characterization is based on the reported notions of strength ("key strength" or "normal strength") and liabilities (for which severe weaknesses are called "key liabilities"). For instance, for the architectural pattern "Pipes and Filters", error handling is a key liability with respect to reliability, while it brings a key strength in terms of portability by enabling filters to be combined. Table I summarizes the knowledge we gathered on architectural patterns and quality attributes. In particular, symbol + stands for strength, ++ for key strength. Symbol – is for liability, – – are for key liability. Symbol = is for a situation of neutrality. Scores have been given as reported in the main study we analyzed. For instance, in the case of Layered it shows a key strength for maintainability. Our lightweight theory encompasses that knowledge as our initial body of knowledge. All the elicited quality attributes, except for Implementability, are present in the Standard ISO/IEC 9126.

In extracting the knowledge specifically related to the Architectural Patterns we observed the following:

- Pattern Interactions. Very little knowledge is provided on how patterns jointly interact with quality attributes.
- Pattern Variants. While pure forms of patterns have been extensively covered by the literature, there is a lack of knowledge and documentation on adopting pattern variants. In order to solve specific problems, patterns need to be tailored. A modified pattern (variant) might better address a quality attribute or in some cases might negatively influence other attributes.

Quality Attribute. Quality Attribute refers to a specific quality model (in our lightweight theory this being the standard ISO/IEC 9126). However, quality models show a lack of coverage. This is because, due to the complex nature of quality attributes, we need to decompose them in sub-characteristics. In our initial theory according the main source, we consider six main quality attributes: functionality, reliability, usability, efficiency, maintainability and portability. However, the authors add a quality attribute outside this reference quality model, namely Implementability. Also, Functionality has been considered only regarding to security sub-attribute. In summary, our lightweight theory adopts the Standard ISO/IEC 9126 as a reference quality model, plus Implementability and Security as quality attributes. The motivation for adopting that quality model is that our initial theory is grounded in [1] and therefore we consider the standard ISO/IEC 9126 as the starting body of knowledge regarding quality attributes, too. Of course, we are aware about the novelties introduced by the revised Standard ISO/IEC 25010; so far, however, our primary studies do not use the revised standard as reference quality model.

Inside this area, another important concept is that quality attributes cannot be realistically considered in isolation but they interact. It is impossible to achieve the same level of quality for all the quality attributes. In this case trade-offs between quality attributes are important and it is necessary to make systematic the associated knowledge.

Interaction. According to the point of view expressed in the main study, we assume that each pattern may impact on quality attributes in a different way. However, we are interested in observing different types of interaction; for instance quality attributes that impact on the patterns by determining variants of them. In some cases, incorporation of tactics in patterns may allow better results with respect to specific quality attributes.

C. Literature Review

In order to identify a set of primary studies on architectural patterns and quality attributes, we performed a SLR following the guidelines proposed in [4]. Our research question is:

- RQ1- What types of interaction exist between architectural patterns and quality attributes?

Examples of "interaction" may include any dependencies like impact, synergies, conflicts, etc. We found 110 primary studies out of 2.942 hits captured by the search string obtained from RQ1. Further details can be found in the full SLR protocol provided as online material (see http://www.s2group.cs.vu.nl/gianantonio-me/).

D. Subset of primary studies

For the preliminary study here presented, we selected a subset of ten primary studies out of the total 110 primary studies. Among them, we included both studies with a wide coverage of patterns and quality attributes and a random set of primary studies. Table II summarizes the main characteristics of this subset.

TABLE II. SUB-SET OF PRIMARY STUDIES

Study Number	**Study Knowledge Elements**		
	Reference	***Patterns***	***Quality Attributes***
1	[5]	Blackboard	Reliability
2	[6]	Agent Pattern	Adaptability
3	[7]	Pipes and Filters	Modifiability and Performance
4	[8]	Repository pattern	Different quality attributes
5	[9]	Five main patterns according to Buschmann	Six quality attributes according to Std ISO/IEC 9126
6	[10]	Layer and Microkernel Pattern/ combinations of patterns	Reliability
7	[11]	Peer to peer patterns	Energy efficiency as quality attribute
8	[12]	Layered	Scalability
9	[13]	MVC variants	Testability
10	[14]	Several patterns	Various

III. ANALYSIS

The analysis has been carried out according to the four conceptual areas identified by coding the main study: Architectural Decision Making, Architectural Pattern, Quality Attribute and Interaction between Architectural Patterns and Quality Attributes. In particular, we are interested in challenging the theory, by describing alignments, discovering new concepts or gaps and detecting mismatching. We call knowledge elements all the information gathered on a specific conceptual area. For instance: a characterization of a specific quality attribute by a scenario or an evaluation on a pattern/attribute interaction.

A. Architectural Decision Making

Most analyzed primary studies (nine out of ten) present some knowledge elements about architectural decision-making. In [9] the aim is to create a structured body of knowledge, based on the combination of architecture candidates (patterns) and quality attributes. That structured knowledge is meant to support better-informed decisions. Studies [5] and [11] highlight the decision of "imposing" a pattern to a system. That decision has a strong impact on quality attributes. Other studies pursue a different approach. They create a theoretical decision model [6] or identify factors for selecting a pattern [14]. Another important aspect emerged from study [8]. This study considers the adoption of solutions (patterns) as a "quality attributes' driven" decision-making process [8]. A different perspective is provided by [7] and [12], by introducing two levels in the decision-making process. Both studies consider two levels in the decision-making process. One level is the solution (pattern) that has been considered as abstract. The concrete system solution adopted is considered as a materialization of that abstraction. The authors considered this materialization as object-oriented solution [7] or a tactic [12]. A similar perspective, but with a different approach, stresses on tailoring patterns, bringing variants of "pure forms" of patterns in the decision-making process [13].

B. Architectural Patterns

In [5] Layered architecture has been considered as supporting and improving flexibility but reducing performance. In turn, Blackboard is considered as an effective solution for the problem presented in the study (a fire alarm system). Blackboard is a satisfying solution for performance and efficiency. Blackboard has been defined as "style" in that study. The authors consider as architectural pattern a Periodic Object pattern. Accordingly, in this study we observe the critical issue of definition regarding styles, architectural patterns and design patterns.

In [6] the fact that patterns can prevent to achieve some quality characteristics has been described. Patterns are the results of the trade-offs among multiple forces (quality attributes): some of quality attributes are achieved; others are hindered by the pattern.

An organized body of knowledge on Pipes & Filters pattern appears in [7]. The authors describe the pattern according the dimension of main features (single-threaded, linear topology and communication protocol), object oriented implementation, advantages and disadvantages.

Advantages and disadvantages are explained by scenarios and relative quality attributes. In this last study, Pipes & Filters is positive with respect to flexibility and performance, but shows some liabilities for scalability and modifiability.

A Repository architectural pattern has been presented in [8]. Repository is described as allowing the exchange of evolving data among software components. In [9] (where the term "style" has been adopted) is possible to recognize "classic patterns" (i.e. patterns as analyzed in [2]) these being Layered, Pipes & Filters, Blackboard, Model-View-Controller and Microkernel. It is interesting that the authors excluded pattern Reflection. The rationale of that choice is that Reflection shows a higher complexity than the others. Complexity has been assessed according to implementability (Reflection is hard to implement). In this study, the authors have analyzed experts' perceptions on two sets of variables (patterns and quality attributes). Experts' knowledge has been standardized by providing information regarding abstract patterns, related problems and solutions. Like in our case, the work in [9] has considered as starting point the work of Buschmann [2]. Study [9] provides many more interesting knowledge elements that we will discuss in section IV. Indeed this study compares patterns and quality attributes among each other, giving some information on which pattern might address specific quality attributes.

Layered patterns are considered crucial for decomposition in [10]. By adopting this pattern, applications are organized in several entities that increase manageability. In study [11] patterns are challenged by a new quality attribute (energy efficiency). New hybrid patterns have been proposed in order to increase the system's energy efficiency. It offers a methodological framework that can integrate or extend our lightweight theory about patterns' variation.

In study [12] there is an analysis at first level of granularity (for instance the layers in a layered architectural pattern). In this study patterns are abstractions of components and their responsibilities. Those abstractions are indicated as useful for architecture recovering. Regarding variants, study [13] offers many knowledge elements on the Model-View-Controller variants. This study shows a great relevance because can be used as a framework for analyzing patterns' families evolution. Finally, study [14] considers patterns selection driven by tactics. Here the authors propose a quantitative method for prioritizing patterns. Quality attributes will be achieved by means of tactics. Table III offers a summary of the most important results of the theory challenged against our literature set.

C. *Quality Attributes*

In general, studies that refer to a quality model have all adopted the standard ISO/IEC 9126 [3]. Mostly, authors keep a high level of generality. So doing, quality attributes are not specified in sub-characteristics. In study [5] several quality attributes have been enlisted (reliability, availability, fault tolerance, performance and robustness). While they do not explicitly refer to any quality model, four techniques have been proposed for assessing quality attributes (scenarios, simulation, mathematical modelling and experience-based reasoning). Accordingly, quality attributes are here indirectly evaluated by using the four techniques proposed.

Study [6] defines quality attributes as non-functional requirements or soft-goals, where the term "soft" highlights the difference to functionalities. Soft-goals represent the forces that a pattern contributes to achieve [6]. Study [7] describes quality attributes by scenarios and they are supported by object abstractions. Scenarios are here considered as object-oriented materialization of pattern abstractions. Study [8] is strongly grounded in Standard ISO/IEC 9126. It focuses on the classification of quality attributes in quality model elements. Quality model elements are characteristics, sub-characteristics, attributes and metrics [8]. Moreover, the quality model is customized according to specific domain-dependent problems. So doing, this study goes beyond the standard ISO/IEC 9126, where there are no guidelines for customization.

The same quality model has been adopted by [9], which however does not specify quality attributes in sub-characteristics. The brought motivation is that, in order to compare quality attributes it is necessary to keep all quality attributes at the same level of granularity. The risk, according the authors, is that a more complex quality attribute might dominate simpler ones. Study [10] characterizes quality attributes by scenarios, without any reference to a specific quality model. Only study [11] presents the peculiarity of introducing a new quality attribute, energy efficiency. This seems to have a relevant impact on the overall knowledge we already have on patterns and quality attributes. Indeed to better achieve energy efficiency hybrid patterns have been proposed. New term "quality feature" appears in study [14]. In this study quality features are enlisted (Availability, Security and Performance) and very-broadly described only by means of scenarios. Finally, study [12] does not refer to any quality model. Here quality attributes are achieved by means of tactics. Accordingly, tactics should be linked to the specific quality attributes they interact with.

D. *Architectural Patterns and Quality Attributes Interaction*

We classified interaction between Patterns and Quality Attributes in two categories: a) Directly addressed and b) Implicitly addressed. In the first category we expect studies that describe, define and analyze that interaction. In the second category we classify all the studies that deal with patterns and quality attributes but do not pay attention to the definition of the interaction.

1) Directly addressed. We have not found any study that aims to describe theoretically and define the interaction between quality attributes and patterns.

2) Implicitly addressed. In [5] the term of "fitness" has been used for describing the interaction between patterns and quality attributes. The appropriateness of a pattern is determined by quality attributes. Quality attributes drive the pattern selection ("imposition" according to [5]). A different idea appears in [6] where patterns are considered as solving multiple forces. Here patterns contribute to achieve several forces (quality attributes), and scenarios describe the way a pattern contributes to achieve a force.

TABLE III. ARCHITECTURAL PATTERNS AND RELATED KNOWLEDGE ELEMENTS

Study Reference	***Knowledge Elements***				
	Patterns	***Quality Attributes***	***Scenario***	***Comparison to the lightweight theory***	***Outcome Explanation***
[5]	Layered	Flexibility	Strength *(+)*	Gap	Flexibility is not present in the *theory* quality model
[5]	Layered	Performance	Liability *(-)*	Alignment	If we consider efficiency as synonymous of performance
[5]	Blackboard	Performance	Strength *(+)*	Mismatching	Problems in determining the difference between performance and efficiency; Blackboard shows liabilities in Efficiency
[5]	Blackboard	Efficiency	Strength *(+)*	Mismatching	*Idem*
[6]	Agent Pattern	Autonomy, Adaptability	Composition of multiple forces	Gap	Agent Patterns, autonomy and adaptability are not present in the initial model
[7]	Pipes & Filters	Flexibility	Strength *(+)*	Gap	Flexibility is not present in the *theory* quality model
[7]	Pipes & Filters	Performance	Strength *(+)*	Mismatching/Alignment	If performance can be considered as Efficiency in the *theory* we have a contradictory interaction
[7]	Pipes & Filters	Scalability	Liability (-)	Gap	Scalability is not present in the *theory* quality model
[7]	Pipes & Filters	Modifiability	Liability (-)	Gap	Modifiability is not present in the quality model of the initial *theory*
[8]	Repository	Unknown	Repository allows to exchange data among software components	Gap	Repository is not a contemplated pattern
[9]	Pipes & Filters	Efficiency	Strength *(+)*	Alignment	Both studies use the same quality model of reference , it can be a validation of the theory because the knowledge in [9] stems from the architects' practice
[9]	Layered	Efficiency	Liability (-)	Alignment	The same of above
[9]	Model-View-Controller	Usability	Key strength *(++)*	Alignment	The same of above
[9]	Pipes & Filters	Usability	Neutral vs Liability (= vs -)	Mismatching	The comparison is difficult in this case
[9]	Microkernel & Layered	Reliability	Strength (+)	Alignment	Can be considered as a kind of validation
[9]	Microkernel	Maintainability	Key strength vs Liability (++ vs -)	Mismatching	This is a point for further analysis
[9]	Layered and Blackboard	Portability	Irregular	Mismatching	This is a point for further analysis
[10]	Layered	Manageability	Layered pattern allows decomposition in smaller entities	Gap	Manageability is not considered in the initial theory quality model
[11]	Hybrid Pattern (Publish-Subscriber plus Client-Server)	Energy Efficiency	Peer to peer patterns are more energy efficient	Gap	In our initial *theory* Energy Efficiency has not been taken in account. Publisher/Subscriber is not considered as a pattern
[12]	Layered (just as example)	For example: scalability	Identification of components and responsibilities addressing quality attributes	New Concept	This study introduces the issue on how abstract into patterns components that address Quality attributes and their responsibilities
[13]	Model-View-*	Testability	Families of Model-View-* patterns; variants of patterns	Gap/New Concept	Variants and testability have not been considered in our initial theory.
[14]	Several patterns	Various	Prioritization of patterns and QAs	New Concept	QAs are achieved by tactics.

In [8], although not explicitly addressed, it is possible to consider the interaction between patterns and quality attributes as response: the adoption of a pattern is the response to a specific quality attribute. The process is quality driven (quality attributes "ask" for a solution, patterns respond by providing the required architecture). Quality attributes continuously shape the solution (refinement) by generating context-dependent variations of patterns. In [9] "fulfilment" (i.e. fulfilling the blend of quality attributes) is the term adopted for describing the interaction between patterns and quality attributes: patterns fulfill a specific set of quality attributes, and there is an explicit interest on quantitative information (i.e. pattern A fulfills quality attribute X, pattern B fulfills more quality attribute Y and how much more? is a research question). In [12] the interaction between patterns and quality attributes is driven by tactics, where quality attributes are concretely achieved by incorporating tactics into patterns. A similar point of view has been developed in [14] where tactics aim to achieve quality. An index calculating the "impact size" determines how much a pattern should change for a tactic to be incorporated in the same pattern [14].

IV. ANALYSIS

In this section we discuss the knowledge elements described in the previous section. In particular, we report on alignment, gaps, new concepts and mismatching, with respect to the four conceptual areas represented in our initial lightweight theory. Our findings have been captured in the revised theory illustrated in Fig. 2, where we emphasized in color red the corresponding new classes or relations.

A. Alignment about decision-making issues

We observed a general alignment regarding Architectural Decision Making. The analysis of the subset of studies confirmed that an in-depth knowledge on patterns and quality attributes should jointly lead to a better decision-making process. However, the analysis also confirmed that it is important to identify the *drivers* of that process (as both interactions and related quality attributes). The suggestion elicited from the analyzed studies is that the theory can be extended with a decision-making model, based on hierarchies for prioritizing quality attributes and select patterns. A decision-making model, as observed in the literature, might be based on a first decision (selecting the pattern according to desired functionalities) and then refine the choice by addressing quality attributes [6]. Refinement can be done by incorporating *tactics*, and/or creating variants of pre-existing patterns. For this reason, we need a stronger grasp on pattern variants. By gathering specific knowledge throughout case studies in pattern variants we can ground decision-making according to the specific context. In this line of reasoning our theory might be considered at abstract level (the pattern solution in a general meaning) and the tailored solution is the variant applied in the system context. Our theory can *support* the process of creating tailored solutions by using concepts like families of patterns [13], i.e. groups of patterns that document an evolutionary process of several variants.

B. Knowledge on architectural patterns

In this section the discussion aims to assess if the interactions patterns-attributes gathered by our initial theory are aligned with the widespread knowledge elements in the literature. As seen in table III we summarize the knowledge elements gathered according to the area of architectural pattern. We observed an irregular landscape: we have equally gaps, alignments and mismatching. Our analysis also uncovers an important gap in our initial theory, namely it does not include any knowledge elements on variants of patterns. The knowledge reported in [13] is a good methodological starting point for extending our theory. However, that study is exclusively dedicated to the variants of Model-View-Controller. Therefore, we need more research in the direction of patterns' variants. Other concepts that might challenge our theory are about tactics implementation in a pattern and the level of granularity that a pattern represents when analyzing an application. Indeed, tactics concretely address quality attributes and represent architectural decisions [12]. According to this idea tactics cover more than one conceptual area of our theory (namely Quality Attribute and Architectural Decision Making). Regarding granularity, it refers to the way of grouping components and responsibilities. At pattern level, for instance, granularity refers to the interdependencies between layers in a layered pattern.

Study [9] presents important knowledge elements regarding patterns and quality attributes: the authors made an effort in comparing patterns as well as comparing trade-offs among quality attributes. In order to *quantify* such comparisons, they gave to architectural patterns and quality attributes a *value* that reflects the perception of practitioners.

Regarding the interaction between architectural patterns and quality attributes, if we refer to table I we observe a substantial overall alignment. In particular, this alignment regards efficiency (strength for Pipes & Filters, liability for Layered), and usability (key strength for Model-View-Controller in both our theory and study [9]). About liability in addressing usability the results are more difficult to compare. For instance, in our theory Pipes & Filters does not address sufficiently usability. In study [9] the value given to usability included by Pipes & Filters is not so different from other values. That value in our theory should be considered as neutral (for instance, usability for Blackboard). For reliability there is a substantial alignment, patterns that have a good impact on reliability are both in our theory and in study [9] namely, Microkernel and Layered. Regarding maintainability we observe a slight mismatching, indeed Pipes & Filters has been considered in study [9] as very good for achieving maintainability. However, in our theory, as reported in table I, there is only a "+" (only a "strength). A mismatching has been detected regarding Microkernel, considered in our theory as having a good impact on maintainability. That pattern has not the same property in study [9] where has been poorly evaluated with respect to maintainability.

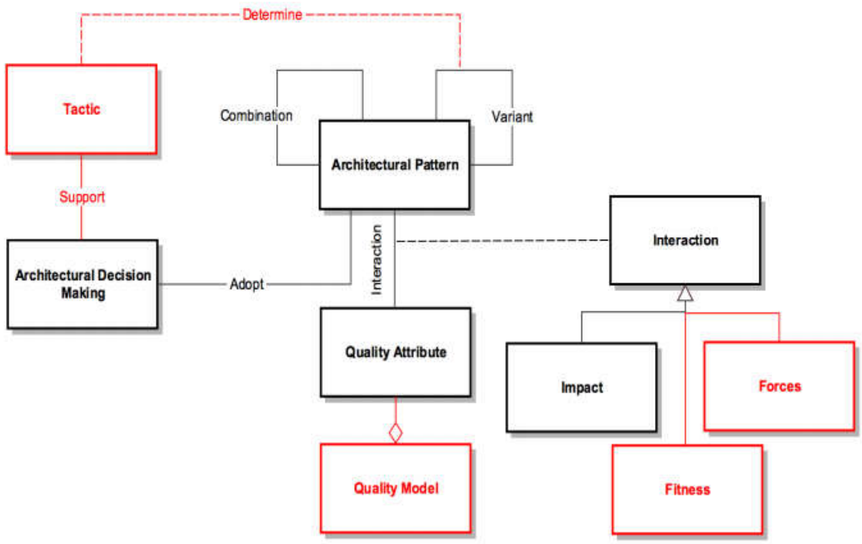

Figure 2-Revisited Lightweight Theory

Portability shows a discrepancy regarding Blackboard (neutral in our theory regarding portability) classified as very poor in addressing portability [9]. Layered in study [9] has been considered as very strong for portability, however in our theory we found just a normal strength. The comparison between [9] and our theory requires further investigation, the approach adopted in [9] (comparing patterns and attributes to each other) might be used for extending the body of knowledge reported in our lightweight theory. Finally, study [12] introduces the issues that patterns are abstraction of identified components and their responsibilities. In this last study patterns are considered as infrastructural constraints.

C. A lack of a unified quality model

Standardization of quality attributes is a key factor to make systematic the knowledge on architectural patterns. Indeed, one of the main obstacles in comparing different studies is the lack of a unified quality model. The standard ISO/IEC 9126 is the only standard quoted in the subset of primary studies. That standard might be considered as unified quality model, as it allows comparison between different cases and allows accumulating knowledge in a systematic way on each pattern/attribute combination. However, the standard ISO 9126 is not sufficient to cover the complexity of quality characteristics. Different definitions for the same quality attribute (for instance efficiency versus performance in two different cases or studies) or new quality attributes (for instance quality attributes that are not present in any other previous quality model) make the comparison between different studies rather complicated. Therefore, the repercussion of a lack of a unified quality model on patterns is a reduced body of reusable knowledge. To this end, an important aspect at this stage, in order to build a theory for supporting pattern adoption, is to encourage the creation of a systematic, detailed and as much as possible unified *quality model*. Updating the quality model to the revised Standard ISO/IEC 25010 should be carefully considered, too.

D. On the interaction between architectural patterns and quality attributes

No studies discuss the interaction between architectural patterns and quality attributes in an explicit way. If that interaction has been implicitly described, we are still unable to conclude if that is relevant from a practical point of view. For instance, it is still undetermined if the term "*fitness*" has the same practical implications of the term "impact" for defining the interaction between patterns and attributes. In general, we can conclude, at least at this stage of analysis, that the interaction between patterns and quality attributes should be driven by quality concerns. In this line of reasoning, quality attributes might be considered as drivers of the entire process of pattern selection and adoption. Again, the relevance of a quality model with a wide underlying body of knowledge appears as a first step to make our theory stronger.

Another point of view on patterns and quality attributes is for instance in study [6]. That study addresses the interaction between patterns and quality attributes in terms of "*forces*" i.e. the help provided by the pattern to achieve some selected quality attributes. Moreover in that study we have a combination of processes: pattern driven (bottom-up) and top-down (for the goals, which are linked to quality). This is an example on how the interaction between patterns and quality is theoretically unexplored, leaving many research questions still open. We consider that interaction is theoretically unexplored because the previous example is likely only one of the wide ranges of possibilities in describing that interaction. In study [12] an important different issue has been introduced.

Firstly more attention has been paid on components and their responsibilities. Components are not always addressing quality attributes: in the specific context analyzed (Architecture recovery) to identify components that address quality is a difficult task. Abstractions to patterns (Infrastructural constraints) and abstractions to tactics (tactics allow quality achievements) are pre-condition for architecture recovery. In this line of reasoning a new concept arises: patterns-quality attributes interaction might be described with the combination of components addressing quality attributes and the tactics related to specific quality attributes.

E. Recap of theory challenges

In this sub-section we group the alignment, gaps, new concepts and mismatching that have been discovered and that pose challenges to our initial theory: The purpose of this section is to outline a preliminary taxonomy of theoretical knowledge elements.

- Alignments and mismatching regarding specific patterns and quality attributes interaction. For instance in table III the case of [5] for Layered (Performance) and Blackboard (Efficiency).
- Gaps in quality model. For instance in table III the case of [11] introducing Energy Efficiency as a new quality attribute.
- Gaps in Patterns catalogue. For instance new patterns like Repository in [8].
- Alignment, Gaps, New Concept or Mismatching in the Architectural Decision Making area. Until now we discovered only alignment and some new concepts for instance systematic decision making models like in [6].
- New concepts that cover one or more than one area: it is possible to evaluate the interaction between patterns and quality attributes with new ideas, for instance the idea of "multiple forces" in [6] regarding quality attributes or the concept of patterns families in [13].

V. CONCLUSION AND FUTURE WORK

In this paper we have presented the preliminary results of a SLR on the interaction between architectural patterns and quality attributes. We analyzed a subset of primary studies using as point of reference a lightweight theory built by analyzing a key literature study. The analysis allowed us to identify alignments, gaps, new concepts and mismatching useful for enhancing the theory. The analysis so far highlights that: 1) there are no studies that directly define the interaction between architectural patterns and quality attributes; 2) some studies define the interaction between patterns and quality attributes with the term "fitness" or using the concept of "forces"; 3) there is a general alignment regarding the architectural decision-making conceptual area; 4) research lacks of a unified quality model. Such model would be useful for making easier and sound comparison among studies and for building a reusable body of knowledge; 5) we observed irregularities (gaps, mismatching and alignment) regarding the patterns strength or liability in addressing specific quality attributes; 6) the theory shows a gap for variants of patterns. The theory might be extended by analyzing variants of patterns through the concept of families of patterns (a documented process and taxonomy of selecting variants); and 7) tactics introduce a new concept for our theory, and cover more than one conceptual area.

In summary, the theory with its main four conceptual areas promises to be an effective and extensible model for creating a systematic body of knowledge on the interaction between architectural patterns and quality attributes. Our next steps are to revise our theory in light of the findings in this work presented, and complete the SLR and integrate the resulting knowledge in a consolidated version of the theory. After that, we plan to challenge the new theory with industrial studies.

ACKNOWLEDGMENT

This work is part of the GINSENG (TIN2015-70259-C2-1-R) project (funded by the Spanish Ministerio de Economía y Competitividad and by FEDER-Fondo Europeo de Desarrollo Regional) and by VILMA (PEII11-0316-2878) and GLOBALIA (PEII-2014-038-P) projects (funded by the Junta de Comunidades de Castilla-La Mancha and by FEDER-Fondo Europeo de Desarrollo Regional).

REFERENCES

[1] N. Harrison and P. Avgeriou, "Leveraging architecture patterns to satisfy quality attributes", In proc. First European Conference on Software Architecture, Madrid, Sept 24-26, 2007, Springer LNCS.

[2] F. Buschmann, R. Meunier, H. Rhonert, P. Sommerlad and M. Stal, "Pattern-Oriented software architecture: A system of patterns". Wiley, West Sussex, England (1996).

[3] International Standards Organization: Information Technology - Software Product Quality- Part 1: Quality Model, ISO/IEC FDIS 9126-1.

[4] B. Kitchenham and S. Charters, "Guidelines for performing systematic literature reviews in software engineering" Keele University and Durham University Joint Report, Tech. Rep. EBSE 2007-001, 2007.

[5] J. Bosch and P. Molin. "Software architecture design: evaluation and transformation." In Engineering of Computer-Based Systems, 1999. Proceedings. ECBS'99. IEEE Conference and Workshop on, pp. 4-10. IEEE, 1999.

[6] M. Weiss, "Pattern-driven design of agent systems: Approach and case study." In Advanced Information Systems Engineering, pp. 711-723. Springer Berlin Heidelberg, 2003.

[7] J.A. Díaz-Pace and M. R. Campo. "ArchMatE: from architectural styles to object-oriented models through exploratory tool support." In ACM SIGPLAN Notices, vol. 40, no. 10, pp. 117-132. ACM,2005.

[8] F. Losavio, N. Levy, P. Davari, and F.Colonna. "Pattern-based architectural design driven by quality properties: a platform to modelscientific calculation." In Software Architecture, pp. 94-112. Springer Berlin Heidelberg, 2005.

[9] M. Svahnberg and C. Wohlin. "An investigation of a method for identifying a software architecture candidate with respect to quality attributes."Empirical Software Engineering 10, no. 2 (2005): 149-181.

[10] D. Bellebia and J. M. Douin. "Applying patterns to build a lightweight middleware for embedded systems." In Proceedings of the 2006 conference on Pattern languages of programs, p. 29. ACM, 2006.

[11] C. Seo, , G. Edwards, S. Malek, and N. Medvidovic. "A framework for estimating the impact of a distributed software system's architectural style on its energy consumption." In Software Architecture, 2008. WICSA 2008. Seventh Working IEEE/IFIP Conference on, pp. 277-280. IEEE, 2008.

[12] F. Solms, "Experiences with using the systematic method for architecture recovery (SyMAR)." In Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference, pp. 170-178. ACM, 2013.

[13] A. Syromiatnikov and D. Weyns. "A journey through the land of Model-View-Design Patterns." In Software Architecture (WICSA), 2014 IEEE/IFIP Conference on, pp. 21-30. IEEE, 2014.

[14] A. Elahi and S. M. Babamir. "Evaluating software architectural styles based on quality features through hierarchical analysis and Fuzzy Integral (FAHP)." In Information and Knowledge Technology (IKT), 2015 7th Conference on, pp. 1-6. IEEE, 2015