Empirical Research for Software Architecture Decision Making
An Analysis

Maryam Razavian
Eindhoven University of Technology
Eindhoven, The Netherlands
m.razavian@tue.nl

Barbara Paech
Heidelberg University
Heidelberg, Germany
paech@informatik.uni-heidelberg.de

Antony Tang
Swinburne University of Technology
Melbourne, Australia
atang@swin.edu.au

Abstract—Context: Despite past empirical research in software architecture decision making, we have not yet systematically studied how to perform such empirical research. Software architecture decision making involves humans, their behavioral issues and practice. As such, research on decision making needs to involve not only engineering but also social science research methods. Objective: This paper studies empirical research on software architecture decision making. We want to understand what research methods have been used to study human decision making in software architecture. Further, we want to provide guidance for future studies. Method: We analyzed research papers on software architecture decision making. We classified the papers according to different sub-dimensions of empirical research design like research logic, research purpose, research methodology and process. We introduce the study focus matrix and the research cycle to capture the focus and the goals of a software architecture decision making study. We identify gaps in current software architecture decision making research according to the classification and discuss open research issues inspired by social science research. Conclusion: We show the variety of research designs and identify gaps with respect to focus and goals. Few papers study decision making behavior in software architecture design. Also these researchers study mostly the process and much less the outcome and the factors influencing decision making. Furthermore, there is a lack of improvements for software architecture decision making and in particular insights into behavior have not led to new practices. The study focus matrix and the research cycle are two new instruments for researchers to position their research clearly. This paper provides a retrospective for the community and an entry point for new researchers to design empirical studies that embrace the human role in software architecture decision making.

Keywords—empirical research, software architecture; decision making, human aspects;

I. INTRODUCTION

Decision making research involves an understanding of how people make decisions. This is a topic that is starting to receive attention in software architecture [1]. Research in decision making is different from research into inventing or evaluating software engineering methods and tools because decision making requires a detailed understanding on how humans think and behave.

Decision making is what software architects do all the time. Software architects are designers who have a high-level view on both business and technical aspects and among other things, they work with many stakeholders [2] and make important design decisions [2]. They make decisions on which architecture style to use, on how to design an API, or what methods should be included in a class etc. However, there are few studies on how software architecture design decisions are made even though all these software architecting activities involve decision making [2]. Decision researchers emphasized that decision making is complex and it is not obvious how decision makers make decisions [3]. Sometimes decision makers themselves cannot tell how they make decisions [4]. We can learn from other disciplines regarding decision making. For instance, classical economics theories assumed that consumers make choices from optimal beliefs and rationale [5], but researchers later found that other forces such as bounded rationality [6] and cognitive biases [7, 8] can influence decision making. The long-held assumption that full market knowledge and rational choices optimize economic decisions does not hold anymore. Consumer rationality and cognitive biases were taken into consideration in economic theories, thereby forming the basis of behavioral economics [5]. Equally, in software design we cannot assume designers to have full knowledge of all design issues and design options to reach an optimal design. The way each individual designer seeks a design solution depends on his or her preferences, beliefs, knowledge and biases. Thus, decision making in software architecture design can also be more complex than what researchers assume. Such a shift in research thinking suggests that software architecture decision making maybe more complex than the traditional software engineering research focus on tools and methods only.

Decision making (DM) in software architecture has scarcely been studied as a human centric activity. In previous work we showed that few research on DM behavior has been conducted in the software architecture field [9]. In the
software architecture community, most interest is on the exploration of areas such as design rationale [10-12], creating software knowledge management methods [13, 14], and creating prescriptive methods and tools [13]. In order to achieve scientific rigor, studies employ empirical research. However, the empirical designs appropriate for researching human DM in software architecture have not been investigated systematically. We do not know which research designs are used for certain types of research.

In the software engineering field, empirical studies are being advocated as a way to validate research results [15, 16] and research methods like experiments and surveys are well established [17, 18]. It was found that seventeen percent of studies in software architecture use empirical research methods such as case studies and experiments and amongst these studies, half of them involve human participants [19]. However, research methods suitable for the study of human behavior in software engineering have not been explored in detail. This subject is challenging because studying human behavior requires not only studying of the technology in use, but also that of the social and cognitive processes that surround the architects’ thinking [20]. Thus, research in empirical software architecture decision making requires the borrowing of concepts and theories from social sciences research. Accordingly, the way empirical studies are designed and conducted can differ from the typical evaluative research in software engineering. For instance, approaches to identify thought patterns are think-aloud protocols and collaborative design conversations [20]. These approaches, however, are seldom used in the software architecture field.

Therefore, to understand and conduct empirical research in architecture DM it is important (a) to understand the research designs of individual studies, (b) to be aware of the insights gained in terms of the focus and objectives of the study, (c) to be aware of research designs and standards in both, software engineering and social science where humans are often the subject of a research.

Several researchers have addressed the challenges of research design in empirical software engineering [15, 16]. To address those challenges many have proposed means for selecting the right research methodologies, and/or provided guidelines for conducting those methodologies. For instance, Wohlin and Aurum [21] provide a decision model to select the best fitting research designs, while Runeson and Höst [22] present guidelines for conducting case studies. Easterbrook et al. [23] discuss guidelines for selecting empirical methods for software engineering which focus on the research methodology. While these works have a general focus on research method and software engineering, we instead zoom into the details of empirical research in the area of software architecture DM. This way we are able to picture focus and objectives in addition to the research designs. Also, contrary to these works, we do not want to give specific guidelines, but provide an overview of and identify current gaps in empirical software architecture DM research.

We ask the following research question (RQ) and the corresponding sub-questions:

RQ: How has empirical research on human aspects of software architecture decision making been done so far and what can we learn from that?

We detail this question into three aspects: the focus of the studies, their objective and the research design. The first two allow us to position the studies according to their insights about software architecture:

RQ1.1: What are the study foci and how to characterize them?
RQ1.2: What are the objectives of the studies and how do they relate to the focus?
RQ1.3: What is the research design of the studies?

To understand the focus of DM research (RQ1.1), we look at the decision making focus and make explicit whether practice or behavior aspects are studied. Decision making practice research is concerned with the process, techniques and tools to aid software decision making, whilst decision making behavior research is concerned with human DM behavior [9]. To understand the details of the focus we look at the data collected with the help of the research data focus. Here we characterize and make explicit whether the study focuses on DM process, outcome or factors. As a result, we provide a new analysis instrument, the Decision Making Study Focus Matrix, to position the research with respect to these two foci.

To identify the objective of DM research (RQ1.2), we look at the design science cycle that embraces three generic study goals: problem investigation (studying DM problems), treatment design (providing solutions), and treatment validation (validation of solutions) [24]. We refine the design science cycle to create another new analysis instrument, the Decision Making Research Cycle which also reflects on the DM focus.

Studying research design (RQ1.3) allows us to understand the variety and the details of how DM research has been conducted. We analyze the strategic, tactical and operational designs as explained in common textbooks [21]. We also consulted general empirical research method books [25].

To answer these research questions we proceeded as follows: in a previous literature study [9], we identified empirical research works in the software architecture discipline that deal with different aspects of human software architecture DM. Using an extended set of literature published until 2017, we analyze the ways DM research has been conducted.
We found the following gaps with respect to the focus and data collection (RQ1.1): there is a lack of papers on DM behavior. Also, the papers study mostly the process of DM and much less the outcome of DM and the factors influencing DM. For the objective (RQ1.2.) we found that the papers study much less treatments to DM problems (few treatment designs and very few treatment validations). No paper studied behavior in the context of a treatment. Thus, there is a lack of improvements on software architecture DM and in particular insights into DM behavior have not led to new practices.

From our analysis, we created two new instruments Decision Making Study Focus Matrix and the Decision Making Research Cycle that can be helpful in positioning and characterizing software architecture DM papers, and identifying gaps. We give examples on how to use them. We encourage researchers to position their research clearly with the help of these two instruments. For the research design (RQ 1.3.) we found basic and inductive research as predominant choices on the strategic level and great variation on tactical and operational designs. It is encouraging that a majority of the papers studied professionals and thus give insights into the software industry. However, often surveys are used. In comparison with the field of social studies we encourage researchers to study DM behavior in more detail by experiments and rely on guidelines developed in this field. This should be followed-up with real world case studies. Furthermore data collection should be refined by think-aloud protocols or auxiliary information such as gestures or sketches to unravel mental models. Overall, we see our work as the basis for a systematic approach for researchers to characterize and design empirical studies that embrace human behavior in software architecture DM.

The paper is structured as follows: in Section II we describe our literature review and analysis procedure and in Section III we introduce our coding schema. We summarize the research results with respect to the focus in Section IV and the objectives in Section V. Strategic, tactical, and operational research designs are discussed in Section VI. In Section VII we compare the results with other fields and we conclude in Section VIII. In the rest of the paper we abbreviate decision making with DM.

II. LITERATURE REVIEW

This research intends to study the human aspects of empirical software architecture DM research. We take a broad view of software architecture that includes software requirements and design. We are looking for literature that studies the human aspects of DM in software architecture design. Human aspects can be about how human decision makers think, communicate, and act; it may also involve the effects on software architecture DM when humans use software decision supporting tools, methods or processes.

Human DM in software architecture is a subject that is rarely studied in software engineering. In order to carry out this study, we contemplated our research methodology, which includes a systematic literature study (SLR) to survey major and relevant journals and conferences [26]. We have decided against the use of automated keyword-based SLR because the use of keyword search can either end up with too much irrelevant literature or too little relevant literature. In the former case, there are many studies on DM in software architecture, but not many of them focus on the human aspects of decision making. Instead, many of the research works focus on the engineering aspects of DM such as methods and processes to model, capture, and reuse decisions. In the latter case, if we use “software architecture” as one of our search phrases, we would miss useful literature such as [S14], [S3] and [S7] that describes software architecture DM. This is because some of these studies do not mention software architecture design even though the activities that they deal with are considered as software architecture activities. Human aspects of DM is a general term that describes what goes on when humans make decisions in software design. DM can be influenced by many human aspects such as cognition, biases, groupthink, communication, and so on. Forming a comprehensive list of search phrases to describe all these related human aspects in DM is challenging. Therefore the basic difficulty is the alignment of search phrases that can reasonably result in a relevant set of literature, and a set in which we can confidently say that it is a representative set of literature. Additionally, we are not interested in the research data of the studies, but instead we focus on the research designs. Thus, search completeness is not so important for us. This is consistent with the practices of looking at research trends and research methods used in the primary studies similar to e.g. Kitchenham [26] who only performed a manual search on selected well-known venues, but did not use automatic searches as the researchers did not synthesize the research data of the selected literature.

Instead of using an automated keyword-based database search strategy, we opted to go directly to specific and targeted conferences and journals that typically publish software architecture literature. This would potentially allow us to find research works on human DM in software architecture where the terms and phrases used in the research works are new and varied, and not yet aligned in the community. Following a manual search of these targeted sources, we used backward snowballing [27] (i.e. using reference list to identify new papers to include) to find any relevant literature to be included in our study. We judge that such a manual and targeted search approach would allow us to find sufficient relevant material as a basis for our research purpose. We did not perform forward snowballing as we
judge that the papers that are in the software architecture field would likely be found anyway as we have searched through relevant sources until the end of 2017. The legitimate finds from forward snowballing in Wohlin’s report represents a very small percentage [27].

In a targeted literature search, we collected software DM research papers that are known to us (Step 1 in Figure 1). From these papers, we identified eight targeted sources that are likely to contain the targeted research works (Step 2). We considered them as the primary sources. These are: (a) Journal of Information and Software Technology (IST); (b) Journal of Design Studies (JDS); (c) Workshop on Sharing and Reusing Architectural Knowledge (SHARK); (d) IEEE Software; (e) Journal of Systems and Software (JSS); (f) Quality of Software Architecture (QoSA); (g) Working IEEE/IFIP Conference on Software Architecture (WICSA); (h) European Conference on Software Architecture (ECSA). Seven of the eight sources are platforms where software architecture researchers often publish their works. JDS is the exception. We picked JDS because it has a focus on design and because there was a special issue with a number of studies of how software designers think [28]. We also considered journals focusing on empirical work, namely Evaluation and Assessment in Software Engineering (EASE) and Empirical Software Engineering Journal (ESE), but did not find any suitable papers between 2005 and 2017.

The researchers collectively and manually searched the past issues of these eight targeted software sources to find relevant papers (Step 3). We looked through all issues for the 11 years from 2005 till 2017. The reason for selecting 2005 is because at that time design rationale study started to take off in the software architecture field with prominent research papers such as [11, 29]. We started this research in early 2016 and hence we finished our literature review at the end of 2017. We retrieved research papers from these primary sources by reading the paper titles and abstracts published by these eight sources (Step 3). In this step, we examined the title and the abstract and looked for key phrases like “decision”, “design decision” or “decision making”. Some of the studies do not have these keywords and so we also look for any evidence that human aspects of DM is involved in each study, e.g. S71 and S35. This step finds potential candidate papers, but it does not guarantee that the paper would be accepted in the study because a paper may not conform to the inclusion criteria such as studying human aspects or conducting empirical research in software architecture. The final inclusion and exclusion of papers are performed after reading the entire paper in Step 6.

There are also secondary sources where we found relevant research papers. With the results from our search in the primary sources and the known papers (Step 4), we used a backward snowballing technique (Step 5) [27, 30] to find more relevant papers from citations. These new papers were then added to the collection of potential articles (Step 4). We also included papers that we knew before this review, some of them are well-known papers dated earlier than 2005 (Step 1). Although some of these papers are earlier than 2005, they also satisfy the criteria defined in Step 6. The papers found from backward snowballing are from Empirical Software Engineering, IEEE Expert, Communications of ACM, ACM Computing Survey, International Journal of Human-Computer Interaction, IEEE Transaction of Software Engineering, Agile Conference, Automated Software Engineering and book chapters (Step 5). We ended up with a preliminary set of eighty-three (83) papers from both sources. During Step 5, we also found twelve (12) research papers that are relevant to software DM, but the subject of these studies is not software development.

Based on this set of papers, we then selected the research papers to be included in our analysis by applying selection criteria (Step 6). First, a selected paper must study one of the two subjects: (a) human factors that affect software DM; (b) human aspects of DM practice involving tools, methods or process in a software development environment. Second, a selected paper must present primary research to yield empirical results. This criterion eliminates papers that are anecdotal or offer secondary research such as high-level literature surveys. Third, if a paper does not relate the research results to software DM in software architecture related activities, the work is excluded from our review. These criteria helped us to eliminate any personal biases in the paper selection.

![Figure 1: Paper Selection Process](Figure1.png)
Each of the four researchers (the authors of [9]) read all selected papers. We arranged the reading, selecting and coding of the papers such that (a) each paper was assigned randomly and read and coded by two researchers; (b) each researcher had to determine if the paper fits the selection criteria; (c) each researcher read at least forty-five (45) papers; (d) a researcher must not code or select the paper s/he wrote. We have finally selected a total of thirty-eight (38) papers. Table 1 summarizes the search results. The columns indicate at which stage the papers were identified and if a paper is selected or not. Step x in signifies that a paper from a particular paper source passes the selection criteria in Step 6. For instance, cell “Step1 in / IST” shows that the known software decision article (Step 1) in IST has been selected after applying the selection criterion in Step 6. Step x out shows papers that do not meet the selection criteria. Si in each cell is the paper identifier. The references of all papers used for the analysis are given in Appendix A.

Table 1: Results of Paper Selection

<table>
<thead>
<tr>
<th>Step 1 – 33 Papers Found</th>
<th>Step 3 – 29 papers Found</th>
<th>Step 5 – 22 papers Found</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 papers selected after Step 6 filtering (20 papers)</td>
<td>Step 1 papers rejected after Step 6 filtering (13 papers)</td>
<td>Step 2 papers selected after Step 6 filtering (14 papers)</td>
</tr>
<tr>
<td>IST S46,S80</td>
<td>S3,S52,S90,S91</td>
<td>S38</td>
</tr>
<tr>
<td>JDS S7,S66</td>
<td>S4</td>
<td></td>
</tr>
<tr>
<td>SHARK S81</td>
<td>S65</td>
<td></td>
</tr>
<tr>
<td>IEEE S/W</td>
<td>S74,S83</td>
<td>S23</td>
</tr>
<tr>
<td>JSS S33,S35</td>
<td>S84,S68</td>
<td>S21,S39,S92</td>
</tr>
<tr>
<td>QoS A S69</td>
<td>S42</td>
<td></td>
</tr>
<tr>
<td>WICSA S30,S32,S47,S60,S67</td>
<td>S29,S37,S85</td>
<td>S19,S26,S61</td>
</tr>
<tr>
<td>ECSA S51,S70</td>
<td>S48</td>
<td>S31,S73</td>
</tr>
<tr>
<td>Others S9,S27,S28,S49,S53</td>
<td>S5,S12,S82</td>
<td>S14,S15,S62</td>
</tr>
</tbody>
</table>

Each paper was coded by two researchers according to the schema explained in Section III. If the codes of the two researchers did not match, we discussed this and adjusted them. An example for such an adjustment is the interpretation of Research Outcome. Our interpretation of basic and applied research differed initially when one of us coded case studies using students as basic research whilst another researcher considered this kind of research as applied research. After some discussion, we defined case studies that involve practitioners as applied research but not case studies involving students.

A. Limitations

In the following we discuss internal and external validity as typical for empirical work. According to Kitchenham et al [31], internal validity tries to minimize bias and to prevent systematic error caused by the design and conduct of the study, while external validity refers to the generalizability and applicability outside of the study. While there are no agreed guidelines on how to mitigate the threats to systematic literature reviews themselves, Kitchenham suggests four quality assessment (QA) criteria for selection of primary studies [26]: QA1. Are the review’s inclusion and exclusion criteria described and appropriate? QA2. Is the literature search likely to have covered all relevant studies? QA3. Did the reviewers assess the quality/validity of the included studies? QA4. Were the basic data/studies adequately described?

Those qualities relate to the extent to which the study minimizes bias and maximizes internal and external validity. Therefore, we use those four quality assessment criteria in the following discussion:

Internal Validity. Considering QA1, we have explicitly defined inclusion and exclusion criteria, to avoid results that depart systematically from human aspects of DM or empirical studies. Given our interest in research designs of the primary studies, we have only included studies that are published in well-known venues. As noted in section II the research data and thus the quality of the data and analysis of the studies are not relevant for us (QA3). Considering QA4 we only analyzed data on research designs that are clearly reported. For instance, we originally wanted to analyze the threats to validity of primary studies, but they were not reported clearly in many of the primary studies, so we left them out of our analysis and results. A threat to the internal validity of literature reviews in general and of our study...
is the data collection. We might have classified the papers wrongly. A particular issue is the full codification of our selected papers. In some cases full codification was either not possible or subject to our interpretation, because the papers do not provide sufficient information. As we interpreted through coding how the research was conducted, there is a chance for internal inconsistency. We mitigated the data collection threat by holding consensus meetings with all four researchers. In the data extraction process each primary study was read by two reviewers. One reviewer acted as the main data extractor, whilst the other reviewer acted as a checker. Any disagreements were discussed in the data extraction consensus meetings. A second threat to internal validity is the general applicability of the coding for characterization and classification. The start-list of coding is extracted from [21] and it has been constantly refined through our coding procedure.

**External Validity** A threat to external validity of literature reviews in general and also of this study is paper selection (see also QA2). The search was organized as a manual search process of a specific set of journals and conference proceedings not an automated search process. This is consistent with the practices of other researchers such as Kitchenham et al. [26] which looked at research trends and research methods used in the primary studies as opposed to synthesizing the data or conclusions of those primary studies. We have restricted the scope to the main software architecture publishing sources and backward snowballing, and we did not use forward snowballing. The threat here is that relevant works from other publication venues may be missing. As such, we are limited in our claim that all relevant software architecture DM literature is included. However, we have surveyed mainstream software architecture research sources that are likely to publish such works. We judge that we have a fair representation of the empirical publications on software architecture DM, and our method is rigorous. As part of the review, we also gathered research papers from other disciplines, notably psychology, cognitive science, and design studies to enhance our understanding of DM in general. We did not conduct any comprehensive literature search in these other disciplines.

### III. CODING SCHEMA

Coding was performed by four researchers (authors of [9]) and every selected paper was coded by two researchers separately. When there was any disagreement between coders, the issue was discussed amongst all four researchers to seek an interpretation of the code as well as the intent of the paper. We sought to eliminate any biases or misinterpretation through this process. During the coding of the papers we used a coding schema (see Table 2). It enables the characterization of each research work based on the four dimensions: **Study Focus**, **Research Cycle Focus**, **Strategic Design**, **Tactical and Operational Design**. The two dimensions, i.e. **Strategic Design** and **Tactical and Operational Design**, stem from the work of Wohlin and Aurum [21]. Although they treat strategic- and tactical- and operational design in one dimension, we decided to divide them into two different dimensions so that we can identify the sub-dimensions and subsequent analysis for each. They address different research design concerns and we wanted to study the interdependencies between them: one addresses the strategic design choices and the other, how those choices are operationalized. In addition, we also considered the dimension of the **Study Focus**, since we wanted to relate the research design of a study with the focus of the study. Furthermore, we considered the dimension of the **Research Cycle Focus** which positions the paper within the design science cycle [24], making explicit the objectives of the study. In the following, we explain each dimension and their sub-dimensions. Table 2 gives an overview of the detailed codes for the dimensions.

As a new dimension we use the **Study Focus** which determines the DM aspects to be studied. We use two sub-dimensions in the **Study Focus** dimension, namely **Decision Making Focus** and **Research Data Focus**.
Table 2: Detailed Coding Schema: The codes are structured into 4 dimensions where we add study focus and research cycle focus to the usual distinction of strategic, tactical and operational design. For each dimension we distinguish several sub-dimensions and for each sub-dimension we used several codes.

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Sub-dimension</th>
<th>Codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Study Focus</td>
<td>Decision Making Focus</td>
<td>DMBehavior, DMPactice</td>
</tr>
<tr>
<td></td>
<td>Research Data Focus</td>
<td>DMFactor, DMProcess, DMOutcome</td>
</tr>
<tr>
<td>Research Cycle Focus</td>
<td>Basic Research for Problem Investigation</td>
<td>Study DMBehavior, Observe DMPactice</td>
</tr>
<tr>
<td></td>
<td>Basic Research for Treatment Design</td>
<td>Treatment Design only DMPactice, Treatment Design relating DMBehavior and DMPactice</td>
</tr>
<tr>
<td></td>
<td>Applied Research for Treatment Validation</td>
<td>Treatment Validation only DMPactice, Treatment Validation relating DMBehavior and DMPactice</td>
</tr>
<tr>
<td>Strategic Design</td>
<td>Research Outcome</td>
<td>Basic research, Applied research, Open Question, Closed Question</td>
</tr>
<tr>
<td></td>
<td>Research Logic</td>
<td>Inductive, Deductive</td>
</tr>
<tr>
<td></td>
<td>Research Purpose</td>
<td>Explanatory, Descriptive, Exploratory, Evaluative</td>
</tr>
<tr>
<td>Tactic and Operational Design</td>
<td>Research Methodology</td>
<td>Real world case study, Limited time case study, Experiment, Survey, Action Research</td>
</tr>
<tr>
<td></td>
<td>Research Process</td>
<td>Qualitative, Quantitative, Mixed</td>
</tr>
<tr>
<td></td>
<td>Participants</td>
<td>Students, Academics, Professionals</td>
</tr>
<tr>
<td></td>
<td>Data Collection</td>
<td>Interviews, Questionnaires, Focus Groups, Observation, Content analysis</td>
</tr>
<tr>
<td></td>
<td>Viewpoint</td>
<td>Retrospective, Current, Just after the Fact</td>
</tr>
</tbody>
</table>

The decision making focus (DMFocus) coding characterizes DM research with regard to the distinction between human DM behavior (DMBehavior) and DM practices in terms of design process, techniques, tools or methods (DMPactice). We already used this distinction when analyzing the content of the papers in our previous literature review [9]. DMBehavior papers study psychological and cognitive aspects of DM and deal with different human thinking aspects. Such studies provide an understanding of human behavior and its potential influence on the ways software engineers make decisions. DMPactice papers either observe DM in practice in general or they study specific processes, methods or tools. DMPactice papers provide insights into the DM of software engineers, or they try new methods to improve DM.

The research data focus (RDFocus) coding characterizes DM research as to which details of the aspect (in terms of data) are studied. Inspired by work of Dorst on decision thinking theory [32], we categorize the DM research data into three categories: (a) decision making factors (abbreviated as DMFactor), (b) decision making process (DMProcess), and (c) decision making outcome (DMOutcome). DMFactors are factors that influence the DM process and constitute the context of the DM process. DMProcess describes the activities during DM such as analyzing the problem or providing arguments for a decision. These activities can be in the mind of the people or visible to the outside through documentation. DMOutcome is the output of DM (e.g. decision documentation).

Note that the two study foci are orthogonal. A study can focus on DMBehavior, and collect data on the factors influencing the behavior (DMFactor), or on how the process is influenced by the behavior (DMProcess), or how the outcome is influenced by the behavior (DMOutcome). Similarly, a study can focus on DMPactice and collect data on how a particular process, method or tool is influenced by factors, or influences the overall process or outcome, or how in general factors influence the process.

We introduce the Decision Making Research Cycle which positions the paper with respect to the design science cycle [24] and the relation of DMPactice and DMBehavior. The design science cycle is a well-established research method in software engineering and is typically decomposed into three tasks, namely: problem investigation (what phenomena must be improved? why?), treatment design (design one or more solutions that could treat the problem), and treatment validation (would these solutions treat the problem in a real world context?). This is called design cycle, because researchers iterate over these tasks many times in a design science research project. Specifically for DM research and building upon the design science cycle we propose the Decision Making Research Cycle. Figure 2 shows the three stages of the Decision Making Research Cycle: Problem investigation, Treatment Design, and Treatment validation.

Problem investigation aims at identifying problems (maybe with a theory) in DM. This is basic research. If we relate that to DMBehavior and DMPactice, there are two possibilities: the basic research studies human behavior that means psychological or cognitive aspects of DM (Study DMBehavior). Or the basic research observes the DM in practice...
(Observe DMPractice). Note that one study could have both codes, if practice is observed including behavior aspects. Based on the insights of the problem investigation a treatment is designed. This is a specific process, method or tool for DM. In stage 2 basic research is performed to understand the aspects of this treatment before it is applied in practice. Here we distinguish again between two cases: either the treatment is studied with relation to behavioral aspects (Treatment Design relating DMBehavior and DMPractice) or not (Treatment Design only DMPractice). For the former ideally insights from the problem investigation with respect to human behavior are taken into account. Finally, in the last stage applied research is performed to validate the treatment in practice. Again we distinguish between the two cases: either the treatment is applied with study of behavioral aspects (Treatment Validation relating DMBehavior and DMPractice) or not (Treatment Validation only DMPractice).

![Decision Making Research Cycle](image)

**Figure 2: Decision Making Research Cycle**

For the **Strategic Design** dimension we adopt the definition of [21] and consider strategy as consisting of **Research Outcome**, the general **Research Logic**, and **Research Purpose**. However, we skip the research approach (positivist, interpretivist, and critical), as this is typically not explicitly mentioned in the papers.

**Research Outcome** distinguishes whether the objective is to conduct basic or applied research. We use the following corresponding codes: **Basic research** (a.k.a fundamental research) creates knowledge about the world, without calling for an improvement of the world [24]. We characterize basic research as focusing on understanding a problem or providing a theory rather than solving a real world problem. For example, basic research investigates how DM is carried out in practice. On the other hand, **applied research** is about improving the world [24]. We characterize applied research as providing a solution to a real world problem by applying knowledge (e.g., a DM practice). For example, applying a certain decision documentation approach to improve knowledge vaporization in a real world software project. Besides basic and applied research, Wieringa [24] also proposes classifying the research outcome by the range of possible answers that is pre-specified. An **open question** contains no specification of its possible answers. A **closed question** contains hypotheses about its possible answers. We adopt this distinction.

**Research Logic** refers to the approach in which research results are reasoned about and derived [21]. Our codes correspond to two common ways of reasoning in empirical software engineering research: **deductive** and **inductive**. Deductive research works from a general theory to a specific outcome and it mainly addresses theory testing. For example, when a researcher collects data to confirm or reject a theory. As such, deductive research works from a general theory (generalities) to a specific outcome (particularities). Inductive research, on the other hand, is based on specific data (particularities) to derive a general theory (generalities). For example, interviews or observations (particularities) are used to propose a broad understanding or theories of DM that are intended to apply beyond the sample of participants interviewed or observed.

The **Research Purpose** refers to the reasons why a research is conducted and can be classified as **exploratory**, **descriptive**, **explanatory**, and **evaluative** [21]. Exploratory research is used when there is not much information available in the topic area and the researcher aims to gather insights into the problem. Descriptive research is used to
describe a phenomenon or the characteristics of a problem. Explanatory research is used to examine the nature of certain relationships between the elements of a problem. Finally, evaluative research aims to determine the impact of methods, tools, or frameworks.

The Tactical and Operational Design operationalizes the research strategy. We adopt the definition of [21] where the tactical level comprises Research Process, Research Methodology, and Participants. The codes are based on the following distinctions: Research Process can be qualitative (focusing on the observations of qualitative data and the interpretation of the data), or it can be quantitative (using statistical means to analyze data) or mixed. For Research Methodology we take a distinction from [22]: surveys, case studies (studying specific real cases, no participation of the researcher), experiments and action research (real case with participation of the researcher). Furthermore, we distinguish two kinds of empirical case studies: real world and limited time case studies. Limited time case studies have a preset duration of only a few hours and work on an artificial case. Only the former provide insights into real practices in software architecture while the latter typically give first insights with students as participants. As there is evidence [20] that for DM it makes a difference whether students, academics, or professionals are studied we also look at the kind of participants.

The operational level in [21] comprises Data Collection and Viewpoint. This includes experiment, survey, and simulation as data collection methods. We view the first two as research methodology and skip simulation. Then we use the following codes: For data collection we use interviews, questionnaires, focus groups, observation, content analysis (also called archival research in [21]). Also, we do not use specific codes for data analysis (such as grounded theory), but only use these codes for the research process which distinguish qualitative, quantitative and mixed analysis. In addition, we distinguish the viewpoint when the data was collected. For the retrospective viewpoint, participants describe DM in past projects, e.g. in surveys. For the current viewpoint, participants are performing DM, e.g. in action research and real world project case studies. For the just after the fact viewpoint, participants describe DM just after they have performed it, e.g. in limited time case studies or experiments. It is preferable to study participants during or just after DM, but often this is not possible so that data is collected retrospectively.

The codes for all papers are given in Appendix B. A short summary of the papers is given in Appendix C. In the following sections we provide the results of the analysis with respect to these dimensions. For each dimension we give the results and then discuss them.

IV. STUDY FOCUS

Research design depends foremost on the focus of the study. To better distinguish between different research foci we introduce the DM Research Focus Matrix (see Table 3) which relates DMFocus (decision making focus) and RDFocus (research data focus). Positioning a research using the matrix, makes explicit whether the focus of human decision behavior or decision making practice are studied, and what kind of research data is collected. Each cell clearly describes an aspect with its details. The cells are illustrated by example research foci formulated in form of research questions. We also show the papers found for each cell. In the following, we first answer the RQ1.1 where we provide an overview of the focus of the current studies, then we discuss what we learned and how the DM Research Focus Matrix can be used in future studies.

A. Overview of the Decision Making Focus

For the DMFocus, we classified the papers in terms of whether they are focusing on human DM behavior (DMBehavior) or DM practices such as design process, techniques, tools or methods (DMPractice). We found 13 papers on DMBehavior, and 25 papers on DMPractice. We show these two categories in Table 4, where we subclassified each category by the main focus of their studies, the number of papers found in each sub-class (i.e. shown within the brackets) and the identified papers in each sub-class. The figures are updated from our previous study in [9].

Decision Making Behaviors. There are 13 DMBehavior papers that studied psychological and human aspects of DM. In this class, the papers deal with different human thinking aspects. We found five papers that study Naturalistic and Rational Decision Making. Two papers dealt with Cognitive Biases. Two papers studied Group Decision Making. Two papers studied cognitive limitation and satisficing behavior, classified as Cognitive Limitations. Finally, two papers studied mental characteristics and experience, and these papers were classified as Mental Representation papers. There is one group where no papers were found. We call it Behavioral Science papers. Behavioral science is one of the psychology areas that are widely studied in management and organizations [4]. There is an awareness and there are studies of behavioral science and DM in the information system field [33]. In our review, we have found no works that investigate organization behaviors, motivations, or personality with respect to architecture DM. The number of behavior science papers shown in Table 4 is zero. Although no such papers were found in the software architecture field, we report this category because other disciplines have shown that behaviors are contributing factors to decision making.
Table 3: The DM Research Focus Matrix to position research with regard to decision making focus and research data focus. Positioning a research on the Matrix makes explicit whether behavior or practice aspects are studied and with which data.

<table>
<thead>
<tr>
<th>RDFocus vs. DMFocus</th>
<th>DMBehavior (13 papers)</th>
<th>DMPractice (25 papers)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DMFactor (9 papers)</td>
<td>Aspect: Collect data on DMFactors to study DMBehavior (3 papers: S52, S91, S93) Examples: Which factors influence a rational/naturalistic DM or group DM? Which cognitive factors (e.g., bias, limitation, mental representation) influence DM in general?</td>
<td>Aspect: Collect data on DMFactors to study DMPractice (6 papers: S14, S15, S26, S51, S73, S77) Examples: Which factors influence the DMProcess in practice? How does the prescribed method/agile development/DM tool/design reasoning/knowledge management influence the factors relevant for the DMProcess in practice and vice versa?</td>
</tr>
<tr>
<td>DMOutcome (9 papers)</td>
<td>Aspect: Collect data on DMOutcome to study DMBehavior (1 paper: S91) Examples: What is the outcome of a rational/naturalistic DMProcess/group DM? Which cognitive factors influence which outcome how?</td>
<td>Aspect: Collect data on DMOutcome to study DMPractice (8 papers: S19, S21, S32, S46, S47, S51, S73, S77) Examples: What is the outcome of the DMProcess in practice? How does the outcome of DM in the context of the prescribed method/agile development/DM tool/design reasoning/knowledge management in practice look like?</td>
</tr>
</tbody>
</table>

Decision Making Practice. We found 25 research papers about DM processes, methods or tools. All of these papers study some aspects of software architecture DM practices. We found six sub-classes. Decision Making Process contains seven papers that investigate the steps software architects take in DM. Four papers on Decision Making Methods investigated how a particular method improves DM. Three papers research Agile Development Method. Agile development methods prescribe steps to facilitate a group of developers to reach goals, schedules, and consensus. We found three papers that describe Decision Making Tools. Four papers describe how Design Reasoning can aid DM. All decisions require some kind of knowledge. Four papers focus on the role of Knowledge Management in DM.

Table 4: Research Papers Classified by DM Behavior and Practice

<table>
<thead>
<tr>
<th>Decision Making Focus Classes and Number of Papers Found</th>
<th>Paper ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>DM Behavior Papers (13 papers)</td>
<td></td>
</tr>
<tr>
<td>Naturalistic and Rational Decision Making – 5 papers</td>
<td>S52, S66, S80, S81, S91</td>
</tr>
<tr>
<td>Cognitive Biases – 2 papers</td>
<td>S53, S93</td>
</tr>
<tr>
<td>Group Decision Making – 2 papers</td>
<td>S3, S60</td>
</tr>
<tr>
<td>Cognitive Limitations – 2 papers</td>
<td>S28, S70</td>
</tr>
<tr>
<td>Mental Representation – 2 papers</td>
<td>S7, S62</td>
</tr>
<tr>
<td>Behavioral Science – 0 papers</td>
<td>Nil</td>
</tr>
<tr>
<td>DM Practice Papers (25 papers)</td>
<td></td>
</tr>
<tr>
<td>Decision Making Process - 7 papers</td>
<td>S26, S27, S51, S61, S73, S90, S94</td>
</tr>
<tr>
<td>Decision Making Methods - 4 papers</td>
<td>S32, S33, S39, S49</td>
</tr>
<tr>
<td>Agile Development Method – 3 papers</td>
<td>S14, S15, S92</td>
</tr>
<tr>
<td>Decision Making Tools – 3 papers</td>
<td>S9, S21, S46</td>
</tr>
<tr>
<td>Design Reasoning - 4 papers</td>
<td>S30, S31, S67, S69</td>
</tr>
<tr>
<td>Knowledge Management - 4 papers</td>
<td>S19, S35, S47, S77</td>
</tr>
</tbody>
</table>
B. Overview of the Research Data Focus

The research data focus characterizes the types of data that depict the focus of the research, i.e. (a) data about the factors that influence DM (DMFactors), (b) data about the activities during DM (DMProcess) (c) data about the outputs of DM (DMOutcome). For the research data focus, we found that 27 papers specifically focus on DMProcess, nine papers focus on DMFactors, and nine papers on DMOutcome (see Table 3). It should be noted that these categories are not exclusive and that there are studies that have more than one DM research data focus and are positioned in more than one category.

C. Combining the Decision Making Research Focus and the Research Data Focus

When we combine the two foci in the DM Research Focus Matrix (Table 3), we see that the research aspects focused are subtly different in each area. We have exemplified these aspects in the Focus Matrix. The Focus Matrix comprises six different generic aspects that can be studied. Table 3 shows that so far DMBehavior studies have mainly focused on one aspect, i.e., collecting data on the DMProcess to study DMBehavior, while DMPractice studies have explored all possible aspects.

The subcategories of DMBehavior and DMPractice (Table 4) give examples for the different research data foci: cognitive bias and cognitive limitations are possible DMFactors, while group decision making is a kind of DMProcess. The mental representation can be a part of the DMOutcome or the DMProcess. The subcategories of DMPractice all give examples of important DMProcess aspects. However, this does not mean that there are only few possible combinations of the two study foci. When we examine the two foci together, we can identify the gaps in the current landscape of DM research:

- **Few papers on DM behavior.** As already described in [9] there is little research on behavioral aspects of DM, and the number is much smaller than in decision practice research. That means that software researchers in general do not pay much attention to the role of human behavior in software DM, and we do not really understand how this impacts the quality of software design.

- **Lack of focus on the DM outcome.** In particular, we found only one behavioral research paper that addressed the final outcome of DM. This means that we do not know how behavioral aspects of DM relate to the actual decisions made or the quality of those. For example, the consequences on decision when group DM is used are unknown. Similarly, the effects of cognitive biases on the outcome are unknown.

- **Lack of focus on factors.** Only three behavioral research papers have looked in detail on the factors that influence DM. As such, we know very little about which factors influence DM and how.

These gaps highlight the lack of important DM knowledge in software architecture and they need to be filled.

D. Using the DM Research Focus Matrix by Researchers

We propose that future research studies position themselves clearly in the cells of the DM Research Focus Matrix. This can help researchers to make the focus of the research explicit and would make it easier for the community to map out related results to understand the current state of research in this area. To position a research project on the matrix a researcher needs to answer the following two main questions:

- Is the study a DMBehavior study, focusing on human decision behavior, or is it a DMPractice study, focusing on a practice, method, or tool?

- Which of the three research data categories are collected: DMFactors influencing DM, DMProcess reflecting the activities of DM, and/or DMOutcome, i.e., the end result of DM?

To exemplify the use of the DM Research Focus Matrix, we present two examples of research with two different mappings on the matrix. Although these two examples cannot represent the whole spectrum of study focus variations, they are useful to show how the different coverage of matrix can characterize different research focuses. The mapping of each example is illustrated in Table 5, and Table 6. The grey cells represent the aspects of DM Research Focus Matrix that are not covered in the corresponding example.
Example 1. Researcher A decides to conduct research on the role of intuition and rationality in DM. First, the Matrix enables researcher A to clearly make the DMFocus of her research explicit: intuition and rationality are cognitive processes of the human mind, this study falls under the DMBehavior research category. Now researcher A needs to detail the research focus and to decide which types of data to pursue (RDFocus). Using the DMFocus Matrix, researcher A has three options with corresponding research questions: (a) focus on DMFactors and study, for instance, which situational factors induce or affect the use of intuition and rationality in software architects as decision makers (RQA1 in Table 5); (b) focus on the DMProcess and study how intuitive and rational decision making are reflected in the architectural DM process (RQA2 in Table 5); and/or (c) focus on the DMOoutcome and study the consequences of intuition and rationality in DM in terms of for instance decision quality or even project success (RQA3 in Table 5). Researcher A could address all three research questions and therefore cover all details of rational or intuitive DM in terms of factors, process, and outcome, or for instance focus on factors only. Likewise, she could cover the three research questions in one single study or in separate studies. Table 5 shows the coverage of the DMFocus Matrix for the research in this example (grey cells represent the aspects that are not covered).

Table 5: DMFocus Matrix for example 1: Focus on covering all details of rational or intuitive DM

<table>
<thead>
<tr>
<th>RDFocus vs. DMFocus</th>
<th>DMBehavior</th>
<th>DMPractice</th>
</tr>
</thead>
<tbody>
<tr>
<td>DMFactor</td>
<td>RQA1. Which situational factors induce or affect use of intuition and rationality in software architecture DM?</td>
<td></td>
</tr>
<tr>
<td>DMProcess</td>
<td>RQA2. How intuitive and rational decision making are reflected in architectural DM?</td>
<td></td>
</tr>
<tr>
<td>DMOoutcome</td>
<td>RQA3. What are the consequences of intuition and rationality in DM in terms of for instance decision quality?</td>
<td></td>
</tr>
</tbody>
</table>

Example 2. Researcher B decides to study design fixation (i.e., cognitive bias) in agile software development. The focus of this research touches upon both DMBehavior (design fixation) and DMPractice (agile software development) aspects. Now researcher B needs to further detail the research focus and decide on the types of data that are pursued (RDFocus). As shown in Table 6 Researcher B could focus on DMFactors and study design fixation as a situational factor affecting DM. Researcher B could first focus on DMBehavior and specifically study design fixation as a cognitive factor in architectural decision making (see RQB1 in Table 6). Based on the results, Researcher B can formulate a hypothesis such as “design fixation leads to sub-optimal decisions” and test this hypothesis in the decision making process (DMProcess) of agile software development projects (see RQB2 in Table 6). As shown in Table 6, the mapping of example 2 traverses the DMBehavior and DMPractice cells. More on this crossover path can be found in Section V.

Table 6: DMFocus Matrix for example 2: Focus on covering both DMBehavior and DMPractice

<table>
<thead>
<tr>
<th>RDFocus vs. DMFocus</th>
<th>DMBehavior</th>
<th>DMPractice</th>
</tr>
</thead>
<tbody>
<tr>
<td>DMFactor</td>
<td>RQB1. Is design fixation a cognitive bias in DM?</td>
<td></td>
</tr>
<tr>
<td>DMProcess</td>
<td>RQB2. If design fixation lead to sub-optimal decisions in agile software development projects, if so, how?</td>
<td></td>
</tr>
<tr>
<td>DMOoutcome</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

V. RESEARCH CYCLE FOCUS

The Decision Making Research Cycle shown in Figure 2 allows positioning DM research with respect to understanding behavior and practice, and for improving software architecture design practice. DMBehavior and DMPractice together provide a picture to improve software architecture DM. In the following, we first answer RQ1.2 and provide an overview of research objectives as we relate those objectives to the DM Research Cycle. Then we discuss what we learned in the studies using the research cycle and how the DM research cycle can be used to help researchers in future studies.
A. Overview of the Research Objectives based on the DM Research Cycle

As can be seen in Table 7, the objectives of DM research have focused on problem investigation (26 papers) and only half of them study human behavior. Fewer papers (10) have provided treatment design and only two have applied those treatments in practice. Behavioral aspects have not been studied in the context of treatments.

<table>
<thead>
<tr>
<th>Research Cycle</th>
<th>Studying or relating to DMBehavior (13 papers)</th>
<th>Observing or only studying DMPractice (25 papers)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Research for Problem investigation (26 papers)</td>
<td>S28,S3,S52,S53,S60,S62,S66,S70,S80,S81,S91,S93 (13 papers)</td>
<td>S14,S15,S26,S27,S30,S31,S35,S51,S67,S73,S77,S92,S94 (13 papers)</td>
</tr>
<tr>
<td>Basic Research for Treatment Design (10 papers)</td>
<td>none</td>
<td>S19,S21,S32,S33,S39,S46,S47,S49,S61,S69 (10 papers)</td>
</tr>
<tr>
<td>Applied research for Treatment Validation (2 papers)</td>
<td>none</td>
<td>S9, S90 (2 paper)</td>
</tr>
</tbody>
</table>

Table 7: Mapping objectives of papers to DM Research Cycle.

From examining the research objectives in Table 7, we identify two main gaps in DM research:

- **Lack of studies on treatments** (that means treatment design and treatment validation). Whilst there are many works that investigate the problems, only a third of the papers have looked at the treatments, and even fewer in treatment validation. Thus, we know little of how we really can influence practice. For software engineering as an applied science, research results in principle should be applicable in practice. The extent to which this could work, however, has been the subject of debate for decades [34], and remains an unsolved problem. Dinar et al. [20] call this the science-practice dichotomy.

- **Lack of studies on behavior in the context of treatments**. The studies, which looked at treatment of the issues identified in the problem investigation, did not study behavioral aspects. Thus, the insights on behavioral aspects of DM in general have not been used to understand the effects of treatments more deeply. In particular this means, that insights into DMBehavior have not led to new practices.

These gaps highlight the lack of important DM knowledge in software architecture and they need to be filled.

B. Using the Decision Making Research Cycle

By using the DM Research Cycle (see Figure 2), researchers in this field can position their work in terms of problem investigation, treatment design, and validation. Such a positioning can help to make explicit the focus of the research in the broader landscape of solving real world DM problems, and transferring the treatments of those problems into real world practice. It would also make it easier for the community to understand typical DM problems and already existing treatments of these problems. To exemplify the use of the DM Research Cycle, we extend example 2 in Section 0 and show an example covering the full research cycle. Figure 3 illustrates the mapping of the example to the DM Research Cycle.

**Example 3.** Researcher B would not only like to study design fixation in agile software development, but also provide a treatment for it. In Problem investigation researcher B can conduct basic research that studies fundamental knowledge of the psychological or cognitive aspects of DM. Thus, a researcher might study “Is fixation a cognitive bias in DM?” (see box 1 in Figure 3). Based on the results, researcher B can formulate a hypothesis such as “design fixation leads to sub-optimal decisions”. To intensify problem investigation in practice, researchers can for example, test this hypothesis in agile software development projects (see box 2 in Figure 3). Next, he could devise a treatment to counteract design fixation or take a specific method he thinks is relevant in that context. Then, he performs basic research relating the treatment with the design fixation. Relating means (a) if/how the fundamental DMBehavior aspects (design fixation) change in the DMPractice context (treatment) and/or (b) if/how the DMBehavior aspects influence the DMPractice processes, methods, or tools. In the example, researcher B can design a de-biasing technique to be used in retrospective meetings in agile projects and perform again basic research with a first experiment (see box 4 in Figure 3). Finally, during Treatment Validation, he applies this technique and learns from its use in real world context. In this example, researcher B may validate the de-biasing solution in a real world agile project. He can, for instance, study how design fixation is different from the one in problem investigation, i.e., is there any improvement in design fixation (DMBehavior) when de-biasing is used in the retrospective meetings of the agile software development (DMPractice) (see box 6 in Figure 3)? These new findings could initiate a new cycle of the DM research. In the new cycle the researcher may go deeper and further study aspects of design fixation which still need...
improvement or he might focus on a new *DMBehavior* aspect such as a mental model. There are also opportunities for synergies where insights in the behavior of one specific research group can be taken into account by other researchers applying specific processes, methods or tools.

**Figure 3. DM Research Cycle in an example, studying design fixation in Agile Software Development (ASD) and proposing de-biasing techniques as a treatment.**

VI. **STRATEGIC, TACTICAL, AND OPERATIONAL RESEARCH DESIGN**

In this section we answer **RQ1.3** and give an overview of the strategic, tactical, and operational research designs (see section III) we found and then we discuss the insights based on it.

**A. Overview of Strategic Research Designs**

In this subsection we show the results with respect to research outcome, purpose and logic.

**Research outcome** is a refinement of the research cycle code. Of the 38 research works, we found that 30 studies are basic research *creating knowledge* and answering *open knowledge questions*; one applied research paper which answers an open question; six studies answer closed questions (i.e., hypothesis testing) as a kind of basic research and one is applied research. In summary, the research in the DM field is predominantly basic research answering open questions (see Figure 4).

**Figure 4. Distribution of Study Classification by Basic / Applied Research with Open / Closed Research Questions**

**Research logic:** We found that inductive reasoning leads the research work in DM as 24 studies use induction and only 14 use deduction (see Figure 5). Note that induction is the prevalent research logic in both *DMBehavior* (10 of 13) and *DMPractice* (14 of 25) papers.
Research purpose: As shown in Figure 6, we found that half of the research work (19) in DM is descriptive research. Exploratory and evaluative research are used in seven and eight studies, respectively. Finally, only four studies carry out exploratory work.

In the following we describe the interdependencies between works with a certain research purpose and their common research outcomes and logic.

Exploratory Research: The seven exploratory studies focus on exploring and understanding the DMProcess and or the DMFactors involved. To create such an understanding, most of these approaches aim to understand the perspectives of practitioners. The main idea is that by observing practitioners or by gaining access to the perspectives of practitioners experiencing different aspects of DM, researchers can also gain access to the problems or characteristics of DM. In sum, exploratory research in DM has created meaning out of what is observed or out of the opinion of practitioners in the field. We found that all exploratory research was basic research. Furthermore, we found that exploratory research always used inductive reasoning. They have inductively created generalizations about the group, process, activity, or situation of DM aspect under study.

Descriptive Research: While exploratory research aims to explore a phenomenon to provide insights for more precise investigation, descriptive research takes the next step and describes a phenomenon or characteristic of a problem. All of the 19 descriptive studies are basic research and the majority creates knowledge of the DMProcess. In terms of research logic, 14 studies used induction to describe DM, whereas only five used deduction to do so. The ones using induction took a broader scope, generalized and then described different aspects of DM. The studies using deduction however, took a narrower scope and looked at DM through the lenses of an existing framework or theory on DM. Those studies go from the general — the theory — to the specific — the results that prove/disprove the theory.

Explanatory Research: Explanatory research’s investigation is deeper than descriptive research in the sense that explanatory research describes phenomena and attempts to explain why a behavior is the way it is. This type of research aims at, for instance, explaining certain ways of DM, or the effects of various factors on DM. All four explanatory studies carried out basic research and focused on the DMProcess, of which three used deductive reasoning and one used inductive reasoning.

Evaluative Research: Evaluative research aims at determining the impact of DM approaches, tools, or practices. The research mainly studied the utility of approaches and tools for software teams, or individuals, and how this utility depends on stakeholder goals. Some research works also evaluated the efficiency and effectiveness of the proposed approaches. Two of the eight evaluative studies were applied research. The majority (6) of the evaluative research uses deductive reasoning and few use induction. In the studies that use deductive reasoning the evaluation is more specific in nature, and typically they are concerned with testing or confirming hypotheses about the effects of use of
a certain approach or tool. The studies that use induction, however, take a broader scope in their evaluations and focus on observing and generalizing the utility of the approach in practice.

Comparing DMBehavior and DMPractice, both are mainly descriptive (8 vs. 11), few exploratory (3 vs. 4) and few explanatory (2 vs. 2). As to be expected, all evaluative research focuses on DMPractice.

B. Overview of Tactical Research Designs

In this subsection we show the results with regard to research methodology and process and the participants. For the research methodology we found that the studies mainly used surveys (3 DMBehavior and 9 DMPractice) and limited time case studies (6 DMBehavior and 5 DMPractice). In addition, we found eight experiments (1 DMBehavior and 7 DMPractice) and seven real world case studies (3 DMBehavior and 4 DMPractice) (see Figure 5a).

Behavioral aspects are rarely researched in surveys. Only one paper does explanatory research using a survey. Two of the DMPractice papers use industrial projects for experiments. This is particularly interesting as it is difficult to determine the impact of a method in a rich context. Real-world case studies are particularly interesting as it is difficult to describe a phenomenon considering the variations in a real world setting.

As shown in Figure 8, the research process was mainly qualitative (7 DMBehavior and 11 DMPractice) and mixed (6 DMBehavior and 13 DMPractice) and only one (DMPractice) study was purely quantitative. Qualitative data is the staple of the social sciences fields. Not surprisingly, in the software architecture DM research where humans play an important role, qualitative data is attractive too.

As shown in Figure 9, eight studies (1 DMBehavior and 7 DMPractice) involved student participants only, two studies (1 DMBehavior and 1 DMPractice) involved a mix of students and professionals. Also, two studies (1 DMBehavior and 1 DMPractice) mixed academics and professionals. The rest of the studies (9 DMBehavior and 19 DMPractice) involved professionals only. Thus, 30 studies give insights into DM in the software industry.
C. Overview of Operational Research Designs

In this subsection, we show the results with respect to data collection and viewpoint. Two thirds of the studies (23) use only one data collection method: In Figure 10, we show the frequency of the combination of methods used. Most often the techniques are used on their own, but also there are many different combinations. Focus group is only used by DMPractice papers, the rest is quite evenly spread between DMBehavior and DMPractice.

As shown in Figure 11, predominantly one viewpoint was used (16 only current, 14 only retrospective and 2 only just after the fact). The other combine the current viewpoint with just after the fact or retrospective view. Much more DMPractice (12 vs. 4) papers use the retrospective viewpoint.
In the following we compare the use of research methodology and data collection over time. This is in particular interesting for the comparison with other disciplines (see section VII). For our analysis we grouped the papers into three periods of roughly ten years (Figure 12 a and b): the first period starting in 1987 until 1997, the second period starting in 1998 and ending in 2007 and the last period starting in 2008. The papers of the first period use as research methodology only case studies, in the middle period also surveys were added. There are much more papers in the last period. They use all kinds of methodologies with a focus on surveys (see Figure 12a). For data collection, first only observation and content analysis were used, then interviews and questionnaires were applied and in the last period focus groups were added and interviews used most often (see Figure 12b). Note that some papers use several research methodologies and data collection methods so that the total number exceeds the total number of papers.

![Figure 12. (a) Research Methodology Used by Period; (b) Data collection Method Used by Period](image)

### D. Discussion of Strategic, Tactical and Operational Research Design

The results show a big variety of research designs. While there is a predominant choice for strategic designs, there is a great variety on the tactical and operational level. In the following, we describe the main designs for the different research methodologies. Collectively, these variations of research design types can be reference points for researchers to select an appropriate research methodology for their own research.

**Variations of limited time case studies:** The research process is qualitative. Analysis is based on predefined codes, e.g. from decision theory or a specific DM approach or general DM activities, or based on open coding. Some studies use quantitative assessment in addition. Tasks can be artificial or real project meetings (only once). The task can be to solve a particular problem or to critique a given solution. The problem is mostly to create a design or an architecture, but can also be a specific activity like effort estimation. Furthermore, the task can involve a specific technique or tool (developed by the researchers) which is being evaluated. Participants come from industry or are students. They work individually or in pairs or larger groups. Data is collected through video, audio, observation, think aloud, notes, and diagrams. The viewpoint is mainly current. A feedback session just after the task can be used. The viewpoint can also be retrospective, if the task is to reflect previous work.

**Variations of real world case studies:** The research process is qualitative. In contrast to limited time case studies the analysis is mainly based on open coding. The studied projects last from 6 weeks to 2 years. Researchers are involved in the projects in the long-term studies. One study involved comparison groups. This is typically only possible with student participants. Similarly to limited time case studies, participants come from industry or are students. They work in groups. Data collection in real-life case studies is typically less detailed than for limited time case studies. The studies used observation, content analysis and interviews or focus groups. The viewpoint is always current with additional interviews current or just after the fact.

**Variations of surveys:** Again, the research process is mainly qualitative or mixed. Data analysis uses mostly encoding, and half of these studies apply quantitative analyses in addition. Very few studies use quantitative analyses only. Half of the studies ask the participants to contribute general experiences, the other half ask the participants to describe specific projects or example decisions. The surveys mainly explore how the participants perform DM, but one survey prescribes a process to be discussed in the interviews. Participants are always from practice and one study involves researchers in addition. Three quarters of the studies involve less than 30 participants, two between 30 and 100, and only one more than 100. The method of sampling is not clearly reported in most cases. One study reports about a wide search through Google and LinkedIn. Three studies mention snowball sampling based on initial personal
contacts. Roughly half of the studies use interviews with a duration between 30 and 150 minutes for data collection, the other half use online questionnaires. The questions are mainly open, sometimes closed, and sometimes mixed. Some studies first conduct a literature search to derive specific categories to be used in the questions or codes. The viewpoint is retrospective, except in one study where the participants are asked to consider given decisions.

Variations of experiments: The research process is mostly mixed. Similar to limited time case studies, mostly experimental tasks are to perform a part of design exercise, while one experiment studies risk analysis. All but one experiment prescribes a method or tool. As expected in experiments, most participants are students. One study involves practitioners only and another study involves both practitioners and researchers. This shows that it is possible to involve practitioners in experimental settings. Data collection methods used are content analysis (decision outcome), questionnaires, and sometimes also observation by the researchers (facilitated by a tool). The current and the just-after the fact viewpoints are combined. In some experiments, participants are asked to evaluate the outcome or report on their approach. Data analysis is always quantitative and with one exception also qualitative as the outcome needs to be coded and the questionnaires include open questions.

VII. LEARNING FROM OTHER FIELDS

In Section IVStudy Focus we found that the number of studies focusing on DMBehavior, i.e. those that focus on the human aspects of DM, is scant. In this section we highlight a number of points for research, with the ultimate goal of inspiring and guiding future research on human and behavioral aspects of software architecture DM. To do so, we compare the status of empirical architectural DM research with similar research in other domains. As mentioned in the introduction we refer to Dinar et al. [20] that gives an overview of empirical studies of design thinking. They highlight the difficulties of data collection and analysis, and emphasize the importance of cognitive studies and empirical research. Our study on architectural DM shares similarities to the design thinking study as both involve human thinking. We think that the insights provided by Dinar et al. [20] and other design studies such as [32] can provide useful hints to software architecture DMBehavior research. In the following we concentrate on research methodology and data collection.

A. Research Methodology for Studying DMBehavior

Studying DMBehavior using experiments. DMBehavior papers study how design DM takes place in people’s minds and what factors affect the designers thinking. As shown in Section VI.B, the majority of these papers use case study designs (both limited time and real world) and only one paper uses experiments. Conversely, history of research in design studies and cognitive psychology [20] has shown the necessity of experiments for establishing causality (e.g., between human factors and design process). Case studies are a great research methodology for identifying a certain observation in its real world context. They are, however, not the best for explaining the thinking behind design. Experiments on the other hand, because they manipulate the independent variables to observe effects on dependent variables, are strong in explaining what causes certain design DM.

Dinar et al. [20] discuss how the empirical studies of design thinking moved from early studies interviewing and observing practice to more controlled studies such as experiments. We found that experiments were used after the year 2008 (see Figure 12(a)), but even more surveys have been used. This indicates in our view that also architectural DM studies should advance to more controlled settings, as in design thinking studies.

Careful design of experiments. Dinar et al. argue that the researchers need to be more careful in the following ways:

First, it is important to know about the factors that play a role in the design activities and their interaction. In DMBehavior research many of those factors are human-related, making the research design of such experiments especially difficult. For instance, there are differences between individuals; no two human subjects have the same background or exact same experience. To control this factor in a subgroup of subjects (e.g., expert vs. novices) a large enough sample size is needed so that the backgrounds get randomized and averaged.

Second, in treatment vs. control groups the results of the experiment should be compared to a base-line such as the control group that did not have the variable (or the manipulating or changing factor). For instance, the control group should follow a “no method” design task. But for some factors finding the control group is difficult. For example, if the factor is the level of experience what would be the control group? We need to define “no experience” in this case.

Third, the design prompt of the experiments (i.e., the specification of the problem to be addressed by the participants) needs to be carefully designed and reported. It is, for instance, important to carefully plan for the complexity of the design prompt as suggested by Pfaff et al. [35]. The reason is that the efficacy of a certain solution is different in complex or simple design problems. To manipulate the complexity of design prompts Pfaff et al., suggest introducing ambiguity or conflicting requirements in the problem representation.

Forth, not only is the experiment design important but also the experiment execution. Many of such experiments involve some sort of intervention (e.g., asking reflecting questions in [36]). The interventions and the way they are applied can play a very important role in the quality of the experiments. For example, the interventions should be
applied in a way that feels natural to the designers involved. The design task is also a critical aspect; it should not be too difficult or too simple for a certain subject or for the dedicated time-span

**Complementing experiments with real world case studies.** In experiments, the confounding variables such as, educational background, experience, or gender need to be controlled. However, controlling confounding variables also constrains the *applicability* of DM methods in real world situations. For instance, design teams are usually composed of designers with different levels of experience and expertise and that cannot be controlled practically. Furthermore, observing the effects of individual manipulated variables in naturalistic situations can be very difficult. Therefore, we argue similar to [20] that DMBehavior papers, studying the role of a human factor on decision tasks, should also consider the complexities in real world settings. Thus, on the one hand more experiments in DMBehavior studies should be performed, but on the other hand those studies need to be followed-up by case studies to ensure the applicability of the results.

**B. Data Collection Methods for Studying DMBehavior**

**Studying DMBehavior using think-aloud protocols.** As seen in Figure 12b, the majority of papers used content analysis, observation, and questionnaires to collect data on DM. We argue, however, that content analysis and questionnaires are not sufficient to reveal the details and complexities of the DMBehavior. The DM of the architects, to a great extent, takes place in their minds. The use of content and questionnaires are inadequate to capture their thinking patterns. For instance, questionnaires can only elicit what architects should do or some levels of agreement and disagreement. The respondents (e.g., the architects) most probably provide the most “acceptable or appropriate answer”. This might only reflect what they found appropriate in DM. If researchers are interested to understand the process involved in DM or the way how the architects work, they need to conduct a more in-depth data collection.

To discover the designers thinking patterns, social sciences studies mainly used think-aloud protocols and protocol analysis [20, 37]. Think-aloud protocols ask designers to verbalize their thinking process, uncovering their thought process. Combined with direct observation they help to study how designers actually perform the design task. Usually such observations are transcribed and codified. While the protocol analysis can be quite powerful, it has limitations as well. Researchers in design studies questioned the effectiveness of think aloud protocols to elicit information about the very design it sought to reveal and argued that aspects of designers thinking such as perception and insight were lost in the verbalization [38]. One way to address this is to annotate the transcribed data with auxiliary information such as gestures and sketches. Another way is to collect rough sketches created by designers to supplement the verbal think aloud data. Many design studies in industrial design or architecture field have accompanied think aloud protocols with design sketches or design drawings (e.g. [39]) to externalize mental models and ideas about design. Likewise, in the software architecture field, simple models of architecture can unravel ideas and mental models in architect’s minds.

**VIII. CONCLUSION**

Decision making is an important activity in design, including software architecture design. DM is a unique human activity involving many aspects such as cognition, behaviors and group interactions. In software architecture DM research, researchers have investigated both behavior and practice aspects of this activity. In this paper we searched eight different research publication sources, between 2005 and 2017, for empirical papers on architectural DM. We also used backward snowballing to find other relevant papers. Using the 38 papers that we found, we investigated four different dimensions of research designs covering 13 different sub-dimensions. The results show that there are predominant choices for the study focus, the research cycle focus and strategic design, whilst the choice for the tactical and operational dimensions is more diverse. To understand the diversity, we looked at dependencies and variations in detail.

We introduced two new instruments to position research on software architecture DM: the DM Research Focus Matrix and the DM Research Cycle. We propose researchers to use these two instruments to articulate the foci and objectives of their work and clarify their study position with a big picture view of the software architecture DM research scene.

With these instruments we identified several gaps in empirical software architecture DM research: There are few studies on behavioral aspects, and in particular we found very few studies that consider the factors and the outcome of DM behavior. This means we know very little about the behavioral aspects of software architecture DM. Furthermore, there are many problem investigation studies, but very few contemplated new treatments or validating those treatments, and none looking at the behavioral aspects in context of new treatments. This implies a lack of insights on possible improvements of software architecture DM. These research gaps need to be filled. We sketch insights from other disciplines on studying behavioral aspects.

In conclusion, our work makes the following contributions (a) researchers can better understand their research design choices and position their works accordingly; (b) given the instruments that we proposed, researchers can map their DM research works, and build their studies on each other’s results for a more coherent progress in the research
on architectural DM. This study has provided the basis of a systematic approach for researchers to design empirical studies that embrace human behavior in software architecture decision making.

**REFERENCES**


Appendix A - Selection of Decision Making Literature


### Table B-1: Abbreviations for the Research Cycle, Strategic, Tactical and Operational Design Coding

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Sub-dimension</th>
<th>Codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Research Cycle Focus</td>
<td>Basic Research for Problem Investigation</td>
<td>Study <em>DMBehavior</em> (ST1a), Observe <em>DMPractice</em> (ST1b)</td>
</tr>
<tr>
<td></td>
<td>Basic Research for Treatment Design</td>
<td>Treatment Design relating <em>DMBehavior</em> and <em>DMPractice</em> (ST2a),</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Treatment Design only <em>DMPractice</em> (ST2b),</td>
</tr>
<tr>
<td></td>
<td>Applied Research for Treatment Validation</td>
<td>Treatment Validation relating <em>DMBehavior</em> and <em>DMPractice</em> (ST3a),</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Treatment Validation only <em>DMPractice</em> (ST3b)</td>
</tr>
<tr>
<td>Strategic Design</td>
<td>Research Outcome</td>
<td>Basic research (b), Applied research (a), Open Question (o),</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Closed Question (cl)</td>
</tr>
<tr>
<td></td>
<td>Research Logic</td>
<td>Inductive (i), Deductive (d)</td>
</tr>
<tr>
<td></td>
<td>Research Purpose</td>
<td>Explanatory (pla), Descriptive (des), Exploratory (plo), Evaluate (eval)</td>
</tr>
<tr>
<td>Tactic and Operational</td>
<td>Research Methodology</td>
<td>Real world case study, (caser) Limited time case study (casel),</td>
</tr>
<tr>
<td>Design</td>
<td>Research Process</td>
<td>Experiment (exp), Survey (sur)</td>
</tr>
<tr>
<td></td>
<td>Participants</td>
<td>Students (s), Academics (a), Profession (p)</td>
</tr>
<tr>
<td></td>
<td>Data Collection</td>
<td>Interview (int), Questionnaires (quest), Focus Groups (group),</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Observation (Obs), Content analysis (content)</td>
</tr>
<tr>
<td></td>
<td>Viewpoint</td>
<td>Retrospective (ret), Current (cur), Just after the Fact (jaf)</td>
</tr>
<tr>
<td>Paper</td>
<td>Year</td>
<td>DM Focus</td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
<td>----------</td>
</tr>
<tr>
<td>S14</td>
<td>2011</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S15</td>
<td>2012</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S19</td>
<td>2008</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S21</td>
<td>2015</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S26</td>
<td>2015</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S27</td>
<td>1988</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S28</td>
<td>1987</td>
<td>DMBehavior</td>
</tr>
<tr>
<td>S3</td>
<td>2012</td>
<td>DMBehavior</td>
</tr>
<tr>
<td>S30</td>
<td>2011</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S31</td>
<td>2010</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S32</td>
<td>2012</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S33</td>
<td>2013</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S35</td>
<td>2011</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S39</td>
<td>2008</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S46</td>
<td>2015</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S47</td>
<td>2012</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S49</td>
<td>1991</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S51</td>
<td>2013</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S52</td>
<td>2012</td>
<td>DMBehavior</td>
</tr>
<tr>
<td>S53</td>
<td>2014</td>
<td>DMBehavior</td>
</tr>
<tr>
<td>S60</td>
<td>2014</td>
<td>DMBehavior</td>
</tr>
<tr>
<td>S61</td>
<td>2015</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S62</td>
<td>1998</td>
<td>DMBehavior</td>
</tr>
<tr>
<td>S66</td>
<td>2010</td>
<td>DMBehavior</td>
</tr>
<tr>
<td>S67</td>
<td>2005</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S69</td>
<td>2008</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S7</td>
<td>2010</td>
<td>DMBehavior</td>
</tr>
<tr>
<td></td>
<td>Year</td>
<td>DMType</td>
</tr>
<tr>
<td>---</td>
<td>------</td>
<td>----------</td>
</tr>
<tr>
<td>S70</td>
<td>2015</td>
<td>DMBehavior</td>
</tr>
<tr>
<td>S73</td>
<td>2013</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S77</td>
<td>2015</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S80</td>
<td>2007</td>
<td>DMBehavior</td>
</tr>
<tr>
<td>S81</td>
<td>2007</td>
<td>DMBehavior</td>
</tr>
<tr>
<td>S9</td>
<td>1991</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S90</td>
<td>2016</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S91</td>
<td>2016</td>
<td>DMBehavior</td>
</tr>
<tr>
<td>S92</td>
<td>2017</td>
<td>DMPractice</td>
</tr>
<tr>
<td>S93</td>
<td>2017</td>
<td>DMBehavior</td>
</tr>
<tr>
<td>S94</td>
<td>2016</td>
<td>DMPractice</td>
</tr>
</tbody>
</table>
APPENDIX C – DETAILS OF DECISION MAKING RESEARCH STUDIES

A. Details of Strategy Level

In the following we describe the strategic research designs of the papers in more detail. We cluster the papers first according to their research purpose and second according to their research data focus.

Exploratory Research

We only found exploratory research for the two categories of DMFactors and DMProcess. It should be noted that no exploratory study focused on the DMOutcome.

DMFactors. S14 and S15 are exploratory studies that focus on the factors influencing DM of 43 practitioners in agile software development. Using inductive reasoning, the researchers generalize decision-making issues according to the experience of the practitioners. This research work is a basic research that creates knowledge about process and issues in agile software development.

DMProcess. S28 is a basic research that creates understanding about the early activities of design as well as the related design issues. The designers are given a lift control case as a design assignment. This research generalizes major issues when designers tackled the lift simulation program based on the video record of their design in action (inductive reasoning). S60 is a basic research that creates knowledge about group decision-making. Using inductive reasoning, the researchers generalize mechanisms and technique used and challenges involved in shared DM based on researchers experience in the software architecture community. S61 aims at exploring how technology solutions are used in architecting. This basic research uses induction and combination of literature study, survey and interviews to come up to such understanding. S70 is a basic research that investigated the extent to which students and professionals look for alternatives in DM. The researchers inductively generalized the DMBehavior of students and professionals and found satisficing behavior in most of the subjects. S67 is a basic research that creates knowledge about how practitioners think about and reason about design decision and design rationale practices. Based on a survey the common types of design rationale were identified using inductive reasoning.

Descriptive Research

We found descriptive research for the three categories of DMFactors, DMProcess and DMOutcome.

DMFactors. S26 is a basic research that creates an understanding about factors in organization. Based on interviews with 25 system analysts, team leads and senior developers S26 inductively elicits 8 factors that influence DM: company size; business factors; organizational factors; technical factors; cultural factors; individual factors; project factors; and decision scope. The main difference between such descriptive research and previously discussed exploratory studies is that exploratory studies aim to explore a decision making phenomenon in general and provide insights for more precise investigation, descriptive research takes the next step and describes a phenomenon or characteristics of a problem. For example, S14 is an explorative study that explores effectiveness of decision-making in agile teams. Participants were asked to discuss topics in importance of decision-making and perception of obstacles. S26, however, is a descriptive research that elicits and describes categories of decision making factor. The purpose of S14 is to generally explore obstacles of decision making in agile teams, whereas S26 aims at describing and characterizing the influence factors of the decision making in the agile teams. S52 describes the challenges of shared decision-making in agile software development teams. This basic research uses induction to generalize challenges from concrete observations. S73 is a basic research characterizing factors of difficulty for architectural design decisions following inductive reasoning. S77 and S51 (S77 is follow-up of S51) carry out basic research that elicits DM influence factors, i.e., personal experiences, requirements and quality attributes, followed by constraints and cost/benefit. These factors are identified inductively based on interviews with practitioners. S91 examines different types of decision-making strategy elements and decision knowledge elements from examining issue reports. S93 studies the different types of cognitive biases in software decision making. From a survey, researchers extract and describe the various biases encountered by the participants.

DMProcess. S3 as a basic research creates an understanding of how software professionals in groups invoke knowledge in their communication, reasoning and DM for software effort estimation. S3 uses observations in 3 effort estimation meetings in industry, which used planning poker and uses inductive reasoning and generalizes ways of invoking knowledge by software professionals. S7 characterize the team behavior in DM and compares it with the behavior of the product designers. This basic research creates understanding about the early design phase of design exploration. To come up to such understanding, S7 uses a descriptive framework of decision-making [40] to deductively analyze and describes the designer’s behaviors. S27 defines and creates understanding about the design process strategies using verbal protocol study of professional software designers (basic research). A cognitive model of software design is used to deductively analyze the observed strategies. S80 defines the relation of Rational DM (RDM) to Naturalistic Decision-Making (NDM). NDM was characterized in terms of decision goal, method, environment and knowledge. This basic research work uses inductive reasoning to generalize the use of decision-making strategies. S81 describes how designers capture decisions and make decisions from different perspectives. This is a basic research that creates understanding about social factors involved in the designer’s DM. Those factors are elicited using inductive reasoning, by generalizing from the qualitative data of 3 case studies. S30 also is a basic research that creates understanding about how they reason in real projects.
Inductive reasoning is used to define designer’s reasoning using a survey. Likewise, S31 is a basic research about naive reasoning for architecture DM of undergraduate students. Inductive reasoning is again used for generalizing and characterizing the naive reasoning. S35 carried out basic research and investigated knowledge sharing between software architects and architecting activities. The study uses survey with architects. To come up to such understanding, S35 uses a framework of architectural activities to deductively analyze and describes the knowledge sharing and other architectural activities. S66 carried out basic research on how software designers explore the problem and solution space and the role of reasoning in DM. This was done inductively by observing designers who were given a task of designing traffic simulator. S51 described software architects during their DMProcess. This is a basic research that inductively generalized the way decisions are made. For instance, local scoped decisions such as a component are typically made by individuals but architectural decisions are made by a team. S92 explores in a case study the failings of decision making in an agile development environment. The research considers decision process, information intelligence used in decision making, and decision quality.

**DMOutcome.** S77 creates an understanding about the categories of decisions and their documentation. This basic research, inductively, extracts such categories based on interviews with practitioners. Categorizing the decisions is carried out according to granularity, scope and impact. Likewise, S51 categorizes decisions into structural decisions and technology decisions. S47 defines patterns for service-based integration based on a systematic literature review. The identified patterns are grouped into four decision levels—architecture, platform, integration and application. Such grouping is carried out inductively.

**Explanatory Research**

We only found explanatory research focusing on DMProcess.

**DMProcess.** S49 explains how capturing Question, Option and Criteria (QOC) can affect the design DM. This is a basic research creating understanding QOC could be useful. The research logic is based on deduction since the analysis is carried out based on the core elements of QOC (Question, Option and Criteria). S53 explains how framing/presentation of requirements affects design process and design outcome. This basic research carried out deductive reasoning by hypothesis testing. S62 study the cognitive characteristics of high software design performers and explains how such characteristics affect the design and design outcome. Here also deductive reasoning was used. S94 explains how reminder cards can help designers less prone to satisficing behavior and improve design reasoning and discourse. The explanations lie in the experimental process that the researchers conducted and in the behaviors of the subjects.

**Evaluative Research**

We found evaluative research for the two categories of DMOutcome and DMProcess. exploratoryFurthermore we capture in detail the qualities that were evaluated for each proposed approach or tool.

**DMOutcome.** S19 introduced valued-based documentation of design rationales. The proposed approach is validated using controlled experiments. The students had to rate the usefulness of the elements of the documentation after making the decision. The researchers evaluated if the proposed approach is feasible and how useful the elements of the documentation after making the decision. S21 proposed a meta-model to capture DM constraints with defined semantics and a collaborative architecture DM approach. The researchers tested the approach and the tool (CoCoADviSE) with controlled experiment. The evaluation assesses the completeness and violations of constraints. They also assess time and effort related efficiency, as well as the effectiveness of users in collaborative DM.

**DMProcess.** S32 suggested the use of descriptive forces viewpoint for architectural decisions. The researchers evaluated the utility of the viewpoint by assessing how it supports the DMProcess. S39 assessed the influence of risk checklists and their role on risk perception and decision-making of software practitioners. S46 reported two other experiments on CoCoADviSE using students. This study assessed the utility of CoCoADviSE by asking what the value of reusable architectural design decisions are. S69 explore the effects of design reasoning on the quality of design by comparing two groups of practitioners. S9 presented gIBIS as a hypertext tool, together with a notation itIBIS. Using a case study, the researchers compared and reviewed how design rationale might make design DM more rigorous and error free. S90 reported an evaluation of a method, GADGET, to increase consensus in group decision making. The research was conducted using students and the researchers found that GADGET clarifies different points of views and increases consensus.

**B. Details of Tactical and Operational Level**

In the following we discuss the papers grouped according to research methodology and sorted according to publication year.

**Limited time case studies**

There are 11 limited time case studies. Most of them give a specific task (typical duration 60-120 minutes) to the participants, capture the behavior and analyze it to understand some aspect of decision-making. The first study is from 1987 (S28) was 8 professional designers were given the lift control problem to design a solution. Their behavior was captured with think aloud protocols, video and diagrams. This was analyzed on the basis of a specific cognitive model wrt knowledge resources. A similar study was done 1991 were 2 pairs of designers jointly worked on the ATM problem in 45 minutes (S49). The task was to critique a given solution. Again video was used for data collection. Here a specific decision-making approach (QOC) was used in the analysis to understand how this approach could be useful. In 1998 the lift control problem was used again in S62. This time 40
professionals selected through peer-nomination were asked to solve the task in 120 minutes. Data collection again involved think-aloud protocols and notes from the participants. The data was analyzed wrt decision-making activities to understand differences between high- and low performers. This involved also quantification. In 2010 3 pairs of designers were given a task (S66). This time it was a traffic simulator. Data was collected through video, but also through feedback session with the pairs. Analysis focused on general design activities, but also used part of a specific approach (SEURAT) to code details. The same material was used in S7 to compare software designers with product designers. In S31 (again in 2010) the task was to design an architecture in 60 minutes. In this study participants were students. They were asked to make design decision explicit. Their behavior was captured by a mind-map and a follow-up interview just after the task. Thus, the viewpoint was cur and jaf.

Besides general decision-making the limited time case study setting is also used to study particular decision situations. S3 (2012) studied 3 effort estimation meetings in industry (2 hour each) which used planning poker. As these were real projects data collection was limited to observation. Analysis looked at selected sequences to understand detailed behavior. S14 (2011) and its follow up S15 (2012) describe a study which used a focus group. 43 participants were acquired during a conference. At first they collected individually examples for design decision and then these were discussed in the group. Open and axial coding based on descriptive decision theory was used for analysis. This study was then followed by mini-case studies in agile teams (S15). These teams had specific obstacles which had been identified in S14 and were selected to be studied in depth. The team experiences were captured through interviews to understand these obstacles. Thus, the viewpoint was ret.

S70 (2015) looked at the amount of design reasoning performed before making a decision. 32 students and 40 professionals participated in the study. The task (presented online) was to create reasons in reaction to 10 design scenarios based on some actual system cases (without time limit). The professional were also required to complete a questionnaire on their approach to the task (with open and closed questions). In the first round the students and 29 professionals completed the task without the questionnaire. In the second round 11 professionals did the task and the questionnaire. The data collected were kind and number of the reasons, the time and the answers to the questionnaire, the viewpoint therefore cur and jaf. Analysis was both qualitative and quantitative.

Real world case studies

There are 6 papers solely about real world case studies and one paper (S81) combining real world case study and survey. We count it mainly as a real world case study.

The first study is from 1991 (S9) where IBIS was used in a two-year field trial during requirements analysis and design. Data collection relied on content analysis. Data analysis is not clearly described. The results include facts like identified design errors as well as overall assessments of the use of IBIS.

The other studies are multiple case-studies. S81 (2007) focuses on social aspects of DM and their influence on tools. Here 3 studies with slightly different focus are reported somewhat shortly. The first study interviewed 25 designers about previous work. The second study comprised 9-10 day observations at 3 different companies. In the third study the researchers participated in design discussions at another company for 6 weeks. The collected data was coded with slightly differing focus.

In S52 (2012) 4 projects from 2 companies were studied to understand challenges of shared decision-making in agile development. The projects lasted 1-3 years and used 2-4 weeks sprints. Data was collected during 11-12 months through observation and content analysis of documentation. Furthermore, all projects participants were interviewed at the beginning and were also involved in discussions after observations. All the data was coded and analyzed to identify the challenges.

S32 and S33 (2012 and 2013) report on a comparative multiple case study where 4 teams of senior students (21 altogether) worked on 4 different projects (duration 20 weeks). Two of the teams (projects) had the additional task to use the decision forces viewpoint for documentation so that the effects of this technique could be assessed. Data collection was quite comprehensive through questionnaire (participants data), work diaries, content analysis of documents and weekly focus groups. The data was analyzed through grounded theory. The codes were then related to predefined response variables characterizing design reasoning. S91 (2016) analyze issue reports from Firefox development. The information was coded 2 trained coders. The coded results were used to identify NDM/RDM decision making. S92 (2017) report on a case study conducted with practitioners through interviews, focus-groups, team meeting observations and document analysis. The data was coded through open coding and axial coding techniques. The analysis focus on decision process, decision intelligence and decision quality.

Surveys

There are 11 papers solely on surveys and one paper (S81 described before) combining real world case study and survey. S67 from 2005 and S80 from 2007 are the earliest ones in our sample.

S67 focused on the use and documentation of architecture design rationale. It analyzed 81 responses from practitioners (selected by availability and snowball sampling) to an online questionnaire. The questionnaire presented different kinds of generic design rationale and asked for a rating wrt. importance, usage, documentation (including barriers and methods and tools) and for additions. The answers used Likert scale enabling a quantitative evaluation. The practitioners answered based on their experiences, but not with a special project in mind.

In S80 25 practitioners (selected by availability and snowball sampling, involving participants of a conference) were interviewed (roughly 45 minutes) about their design DM, in particular cues, knowledge, options, experience, time pressure and external goals.
The answers were coded and analyzed through content analysis and explanation building in order to derive a model on DM in software design. The interviews were structured according to the critical decision method. The practitioners reported about specific experiences, but it is not clear, whether this focused on a particular project.

The other surveys were conducted between 2011 and 2015. 

**S30** (2011) presents a descriptive survey with 53 answers on architectural DM, in particular architectural analysis (problem space), architectural synthesis (solution proposal) and architectural evaluation (solution selection). The participants were selected from industry by snowball sampling. They were asked to think of one specific project. As for S67 an online questionnaire with mostly Likert scale and quantitative analysis was used. Two questions were open and the answers were coded.

**S35** (2011) reports on a large-scale survey with 142 answers to an online questionnaire focusing on knowledge sharing between architects. The participants were recruited from 4 companies. As input to the study a model of architectural activities and support methods for knowledge sharing was used. The questionnaire was very carefully designed with a pilot study and statistical analysis. It presented items and asked for agreement on a Likert scale based on the general experience of the participants. Also the analysis used several statistical methods. The results were sent to the participants asking for general feedback.

The interviews (100-150 minutes) of **S47** (2012) involved 9 architects (convenience sample?) with the idea to evaluate a set of architectural patterns for platform integration. These patterns were later also applied in a research project with industry. The questions asked for an explicit architectural assessment of an application and relations to prior DMProcesses. Thus the viewpoint was cur and ret. The qualitative data analysis is not described in detail.

**S51** (2013) is based on (30-60 minutes) interviews with 9 experts from 6 companies. It aims to understand the kinds and influences on design decisions as well as the process and documentation. Questions were open and then coded based on predefined categories corresponding to the research questions. The participants reported general experiences (not a specific project), but were asked to give examples.

**S73** (2013) reports on a questionnaire about characteristics and difficulties of architectural decisions with 43 answers from 23 countries on 5 continents. The participants were acquired through LinkedIn and Google. As for S51 participants were asked to give examples for good and bad decisions and then to describe them in detail. This included the assessment (Likert scale) of 22 statements on the decision and the DMP. Thus, qualitative and quantitative analysis was applied. The participants were grouped into junior and senior architects and their answers compared. Similarly, assessments for good and bad decisions were compared.

**S60** (2014) distributed an online questionnaire on group-DM (practices, techniques and challenges) to 23 practitioners and 7 researchers involved in industrial projects. The participants answered based on general experience. The questions were partly multiple-choice and partly open leading to both qualitative and quantitative analysis. The answers were clustered and summaries derived for clusters.

**S77** (2015) reports on (30-80 minutes) interviews with 25 experts from 22 companies in 10 countries. The open questions focused (similar to S51) on design decisions (kinds, documentation and influences). Participants reported based on general experience. Codes were derived from the research questions and then refined during coding. Analysis used structuring and summarization.

**S26** (2015) re-analyzed the interviews of S77 wrt the decision-making process

**S61** (2015) focuses on technology decisions. 7 experts were selected. (60-120 minutes) Interviews were held two days where first experiences were asked to give examples regarding several concepts derived from literature. On the second day the refined concepts were discussed and rated by the participants. The answers of the first day were analyzed to refine the concepts and are reported as quotes. The ratings of the second day were quantitatively analyzed.

**Experiments**

There are 8 experiment studies, 3 in 2008 and 3 in 2014 or 2015, and 2 in 2016.

**S19** (2008) reports on an experiment to evaluate the feasibility of a tailored design decision rational documentation. The experiment project was a public transportation system. 50 master students were assigned different roles and decisions and had to perform different scenarios for use of the documentation. The students had to rate the usefulness of the elements of the documentation after making the decision. This rating was collected and analyzed quantitatively. Thus, the viewpoint is jaf.

**S39** (2008) reports a role playing experiment with 128 software practitioners from 4 companies to study the influence of checklists and roles on risk perception and DM. The experiment project was an online bank system development. There were 4 treatments groups (with or without role resp checklist). Through a website participants were confronted with project scenarios and asked to assess risks and decide about continuation. Thus, the data collection viewpoint was cur. The risks determined without checklist were coded. Thus, data analysis was quantitative and qualitative.

**S69** (2008) reports on an experiment on the effects of design reasoning on design quality. 20 designers from industry and academia were asked to design a user interface, one group with and the other group without a predefined design reasoning process. The experiment project was a monitoring system for cars developed in industry. Data was collected based on think-aloud protocols during design and retrospective think aloud protocols on the derived designs by the participants, as well as observation and rating of the design through the researchers. Also time was measured and the interviews after the task included closed question on design quality. Thus, data collection viewpoint was both cur and jaf and both qualitative and quantitative analysis was applied.
S53 (2014) involved 42 post-graduate students in a randomized control experiment to study the influence of the framing requirements vs. ideas on design creativity. The experiment project was an existing health-related App. The participants were asked to write down designs. The time given was 60 minutes. Also a post-task questionnaire was applied. Thus data collection was based on content (cur by the researchers) and interview (jaf by the participants) and data analysis qualitative and quantitative.

S21 (2015) investigated a tool enforcing constraints on collaborative architectural DM. 48 students participated in a practical exercise (two sessions of 90 minutes) to design a service-based system. All students worked with the tool which also provided architectural patterns, but in the control group the functionality for enforcing collaboration constraints was hidden. Only a textual description of the constraints was supplied. The data collected consisted of the actions executed (logged by the tool), the time, the completeness and the violations. Thus, analysis is qualitative (to assess completeness and violations) and quantitative.

S46 (2015) tested the effects of reusable architectural knowledge on DM in two controlled experiments using the same tool as S53. In the first experiment 49 bachelor students were involved, in the second 122. All students worked in a 90 minute session with the tool which provided patterns, but the experiment group also had access to reusable decision models in the tool. In the 2 experiments different projects were used. The data collected consisted of the actions, the time, the number and quality of decisions. Thus, the viewpoint is cur and analysis comprises quantitative and qualitative ratings.

S90 (2016) tested the effects of GADGET on consensus in architectural decision making. The researchers did an exploratory study to study the practical applicability of GADGET in the industry, then conducted an experiment with 113 students to evaluate the effect with and without GADGET on decision consensus.

S94 (2016) tested the effects of a reminder card game of design reasoning. The experiment involved experimental and control groups of students. These students were given the same design exercise, and the experimental group would use the prescribed cards during their design session. The design dialogue was captured and analyzed to see whether the groups equipped with the reminder cards reason more during their design.