Describing and Communicating Software Architecture in Practice: Observations on Stakeholders and Rationale

Kari Smolander[1] and Tero Päivärinta[2]

[1] Lappeenranta University of Technology, Telecom Business Research Center
P.O. Box 20, 53851 Lappeenranta, Finland
kari.smolander@lut.fi
[2] Agder University College, Dept. of Information Systems
Serviceboks 422,4604 Kristiansand, Norway
tero.paivarinta@hia.no

Abstract. Despite considerable attention paid on software architecture, the organizational aspects of architecture design remain largely unexplored. This study analyses the stakeholders participating in architecture design in three software companies, their problems in relation to architecture, and the rationale for architecture description they emphasize. This qualitative, grounded-theory-based, study shows how the stakeholders' rationales for describing architecture exceed the plain programming-in-the-large metaphor, emphasizing such issues as organizational communication, and knowledge creation and management. Whereas designers alone highlighted architecture as the basis for further design and implementation, the other stakeholders emphasized architecture mostly as a means for communication, interpretation, and decision-making. The results suggest a need for further research on practices and tools for effective communication and collaboration among the varying stakeholders of the architecture design process.

1 Introduction

Software technologies have been under enormous change. The complexity of software and information systems has exploded in several domains. The trends include the diversification of user interfaces, the distribution of processing and data, and the utilization of mobile and Internet technologies; emphasizing the issue of architecture in the software engineering research as well as in practice.

Constructive and theoretical research approaches have dominated the software architecture research. The constructions of numerous architecture description languages (ADLs [1]) represent an example of this. The ADL school pursues "right" and "exact" high-level structures of software architecture, which are inspected and evaluated according to the requirements and then, ideally, generated to executable artifacts [e.g. 2, 3]. Architecture is thus regarded mainly, often plainly, as a means for further design

A. Banks Pidduck et al. (Eds.): CAISE 2002, LNCS 2348, pp. 117-133, 2002.
© Springer-Verlag Berlin Heidelberg 2002

and implementation, "programming-in-the-large" [4], from the viewpoint of the "system designer" [5-8].

Other possible rationale for architecture, such as organizational knowledge creation or achievement of mutual understanding among diverse stakeholders of software development, are rarely discussed in the research reports or it is just assumed that these are in-line with the programming-in-the-large, which represents the primary challenge. Among the rare reports discussing some organizational aspects related to architecture, Kazman et al. [9, 10] have implicitly recognized the existence of different stakeholders in actual software-producing organizations. However, they regard other stakeholders than software architects only as informants for the generic architecture design process instead of active stakeholders with the needs and purposes for architecture of their own. Bosch [11] presents a case study on the possibilities for organizing software product lines. Still, he discusses little about the rationale and needs of different stakeholders in architecture design and description. Robbins and Redmiles [12] consider the idea of diverse knowledge and the theory of reflection-in-action [13], but focus on the design work of a single architect leaving other possible stakeholders and their interests untouched. Grinter's [14] qualitative study covering 17 architects in a telecom corporation highlights the communicational and political skills required from architects, as they must communicate with (and influence to) other stakeholders in numerous situations. Still, the role of the other stakeholders remains vaguely discussed.

Our paper attempts to delve into the rationale of architecture design and description in practice, taking the roles of diverging stakeholders into account. Given that the structures of software-producing organizations vary, software products vary, and business strategies and practices vary, does the rationale for designing and describing architectures vary? Which purposes of architecture design and description emerge in practice? Is practical architecture design and description guided only by the needs for technical design and implementation from the viewpoint of the architects and designers? The state-of-the-art reports discuss little about these questions [15], and the other than the technical purposes of architecture are often referred to with ad-hoc lists [16-18] without explicit empirical grounding or verification of their importance.

In three software-producing organizations, we examined the participating stakeholders, the problems they face in relation to architecture, the rationale for architecture description they emphasize, and the architectural viewpoints they use when dealing with architecture. We hypothesized that the rationale for describing architecture exceeds the plain programming-in-the-large metaphor, covering also such issues as organizational and inter-organizational communication, and knowledge creation and management. Especially, we wished to focus on the combinations and relationships between the rationale, stakeholders, and viewpoints used in the architecture design and description process.

Architecture descriptions cover the documents, presentations, plans, sketches, and other genres of communication that are used in the architecture design and utilization. This represents an idea somewhat broader from that of the IEEE 1471 recommended practice, which emphasizes the architecture's documentation [16]. A stakeholder is a person or a group of people involved in creating or using architecture descriptions in a way or another: e.g. designer, project manager, architect, general management, customer. The repertoire of stakeholders varies between individual organizations.

A rationale is an underlying reason for creating architecture descriptions. In the IEEE recommended practice [16], a viewpoint is a specification from which individual views are developed by establishing the purposes and audience for the views and the techniques for their creation and analysis. Instead of necessitating any existing specification to identify a viewpoint in our analysis as such, a view and its specifying viewpoint were not explicitly defined beforehand, as we hypothesized that the organizations likely had shortcomings in such definitions, and our prejudice on the viewpoint definition might have constrained the data collection and analysis.

The paper is structured as follows. Section 2 explains the qualitative research process that took place in three software-producing organizations. Section 3 describes the key stakeholders of architecture design in the target organizations accompanied with their emerging problems and emphasized rationale, and a comparison of the differences and similarities identified among the organizations. Section 4 discusses implications for practice and research. Finally, Section 5 ends the paper with conclusions.

2 Research Process

We had access to three software-producing companies, hereinafter referred to as Alfa, Beta, and Gamma (Table 1), which had recognized the importance of architectural design for several years. Their parallel analysis and comparison was particularly interesting for our research purpose, as the nature of the organizations' products and customer relationships varied.

The research followed the grounded theory method, a research approach for creating and elaborating new theory from qualitative data to be collected within the research theme in question [19]. Qualitative approaches have been regarded as useful starting points in those fields whose literature base is in its infancy [20] – as is the case of research on software architecture design in real-life organizations. The grounded theory approach can also provide new insight into the already accepted theories, even "paradigms", of the research field in question [19] – as is the case concerning the programming-in-the-large metaphor in the software architecture research.

Table 1. Three target organizations

	Alfa	Beta	Gamma
Size (people in sw process)	200	200	400 + 600 (in two divisions)
Typical products	Embedded software products (within a certain technology) for mobile terminals	Software-based telecom-services and service platforms (running on servers)	Tailored information systems (also mobile and web-based) and services
Typical customers	Manufacturers of mobile hardware	In-house customers (providers of commercial telecom services) within the corporation	Project-by-project varying customers (telecom operators, public administration…)
Interviewed stakeholders	1 architect, 1 designer, 4 managers (1 project manager)	2 architects, 2 designers, 2 managers, 1 manager of an in-house customer	3 architects, 1 designer, 2 managers

Two researchers designed the data collection and analysis process collaboratively. The first author had conducted preliminary discussions with the intermediaries of each target organization in spring 2001. Those discussions had revealed the fuzziness of the concept of software architecture in each of those organizations, making them eager to participate in this research so that the situation would be clarified also for their development efforts, in addition to our "pure" research goals. The first author conducted altogether 19 interviews of the key stakeholders named by the organizations' intermediaries (Table 1) during August and September 2001. The semi-structured interviews covered the themes and sub-themes listed in Table 2. In addition, the intermediaries and interviewees provided documents about the organizations' existing software process specifications and examples of actual architectural descriptions. The role of the second author, an outsider from practical data collection, aimed at more "distance" compared to the first author, to facilitate data analysis [21].

Table 2. Interview themes and sub-themes

Theme	Subquestions
Role and tasks	Your role in relation to software development? What kind of products are you developing? How is their architecture described?
Need of architecture descriptions	Do you need architecture descriptions in your work? What kind? Why and for what purpose? Do you produce architecture descriptions in your work? What kind? Why and for what purpose?
Role of other stakeholders	What other parties or stakeholders need and produce descriptions? For what purpose? Explain the role of each stakeholder in detail.
History and experiences	How long has architecture been described like this? Do you remember any changes in the way of doing? Why there have been changes?
Change management	Will there be changes to architectural descriptions during the system's lifecycle? Who will update the descriptions?
Description practices	What is missing from the current architectural descriptions and what would you need more? Do you think that architectural descriptions should be improved? Why? Do you have any ideas about that?
Tools and languages	What kinds of tools for architecture design are available in your organization? What kinds of deficiencies do they have? What kinds of tools would you need in addition to the present tools when describing architecture? Why? How would you improve the situation? What kinds of description languages are needed for architecture description? What do you need to describe with these languages? Why?

The basic idea of the grounded-theory-based data analysis resides in finding conceptual categories and abstractions related to the research goal from a rich set of interviewees' mentions and other data, and in combining these categories meaningfully to provide theoretical insight into the phenomenon in question [22]. For the analysis, the interviews were tape-recorded and transcribed, resulting in c. 400 pages of text. A software tool designed for grounded-theory-based data analysis (ATLAS.ti™) was used for managing and analyzing the data, including also the documents about software processes and the examples of architecture descriptions. We started the analysis simply by searching for mentions about different stakeholders related to the architecture description process – we also had decided beforehand to focus on the mentions

about the rationale of those stakeholders for producing or using architecture descriptions. The conceptual categories of 'stakeholder' and 'rationale' can thus be regarded as the high-level "seed categories" [23] of the data analysis. This "open coding" phase proceeded iteratively in parallel with the "axial coding" phase, in which relationships between the identified categories were built [19]. The first author conducted the initial coding iterations, after which the categorizations and interpretations were discussed and re-checked against the data collaboratively between the both authors.

Table 3. Identified stakeholder roles

- Customer
  - Business responsible
  - Technology responsible
- Designer
  - Chief designer
- Project manager
  - Technical project manager
- Architect
  - Chief architect
  - Project architect
- Customer management & marketing
  - Account manager
  - Salesman
- General management
  - Project steering group
  - Department manager
  - Team leader
- System analyst
  - Product engineer
- Data administration
  - Internal data administration
  - Customer's data administration
- Product management
- Other projects & organization
- Other suppliers
  - Subcontractors
  - Suppliers of connecting systems
  - Hardware vendors
- User
  - Known users
  - Anonymous users
- Testing & quality assurance
  - Tester
  - Quality manager
- UI Designer
  - Technical designer
  - User experience team
- Production organization
  - At customer's site
  - At vendor's site
  - 3rd party service
- Support
- Technology management
- Authorities
- Process development
- Hardware integrator
- Documentation specialist
- Consultant

3 Stakeholders of Architecture Design and Description

3.1 Stakeholder Roles in Three Organizations

More than 20 stakeholder roles altogether (Table 3) were mentioned in the data. Each of them participated in the architecture design and description process, either by designing or describing architecture or by using the descriptions. The general-level stakeholders that carried similar connotations of their roles in the target organizations according to our interpretation are shown at the first level of indentation. The second level of indentation contains some variance inside the conceptualized stakeholder role in question (is-a relationship). In practice, an individual can play several roles. For example, a person affiliated as a 'designer' may operate also in the product support, as an architect in small projects, or even as a project manager. People assigned to the general management can be (and usually are) strongly involved in the customer management and marketing alike.

In the following, we describe the stakeholder roles in each target organization. Those roles that were mentioned more than once in the data were included in our maps describing the occurrences of the stakeholders in individual organizations.

3.1.1 Alfa

Figure 1 depicts the identified stakeholder roles in Alfa. The boxes represent a stakeholder role or a wider organizational entity that includes roles. A stakeholder role located inside a box operates within that organizational entity: for example, a project manager typically operates within a project and the project within a development organization. In Alfa (as well as in Beta and Gamma), a project designs and produces the software, including its architecture. A set of designers implements the software, and a project manager coordinates the project. Hardware integrators integrate the software to the constraining hardware. Three roles overlap the project border. The user experience team and quality management operate, in part, independently from the projects. The role of an architect was evolving at the time of the interviews. The number and resources of dedicated architects appeared to be very limited, and due to that fact, the architects operated rather independently from the projects and vice versa.

Alfa's management had a strong technology orientation. Four management roles and the role of process development were identified in the context of architecture design and description (Figure 1). Rapidly evolving mobile technologies and Alfa's strategic commitment to a certain technology necessitated that the general management took a number of conscious and serious risks related to the product's architecture. Moreover, as Alfa's business was rather product-based, the product management guided the architecture design and description process as well.

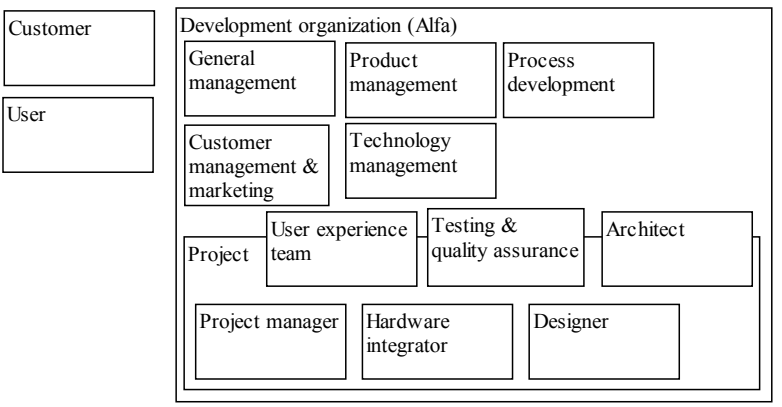

Fig. 1. Map of identified stakeholder roles in Alfa

The external stakeholders played less important roles compared to Beta and Gamma. The customer was definitely a key stakeholder of architecture design, although their relationship to projects, including designers, project managers, and architects, was rather distant. One explanation resides in the fact that Alfa operated as a subcontractor delivering specified product components based on formal documents provided by the customer, and Alfa's customers did not always want to share their technical knowledge with Alfa for competitive reasons, therefore limiting their interaction. Alfa thus developed its products and components quite independently from its

customers. The actual software users were mostly anonymous from the viewpoint of Alfa's internal stakeholders.

3.1.2 Beta

In Beta's project-centric development organization (Figure 2), a project includes a manager, a product engineer, or a system analyst specifying requirements, designers, and a user interface (UI) designer and a documentation specialist in lesser roles. A project can include also a so-called project architect working at a level more detailed than the chief architect, who carries the responsibility for general-level solutions and policies. The product support and quality management are partly organized according to projects. No formal, fully project-independent support organization existed, as the customer operated within the borderlines of the same corporation, and the customer organized the support. The general management of Beta was somewhat technically oriented, being involved in architecture design and description. The management had initiated several development programs for enhancing design practices.

Beta, as an in-house developer, interacts intensively with the customers, who define the business, coordinating the relationships to external stakeholders, such as telecom service users, external customers, and telecom authorities. The subcontractor plays another role of external stakeholder. Based on the process documents, Beta was supposed to interact directly with subcontractors. In the interviews, however, no subcontractors were mentioned. Customers own the software products developed by Beta. The software development projects thus took place typically in parallel with the customer's product development initiatives, including business and concept development for telecom services. In addition to product development, customers run the software (production organization) and interfaces to the billing systems and other operative information systems (data administration). (Figure 2)

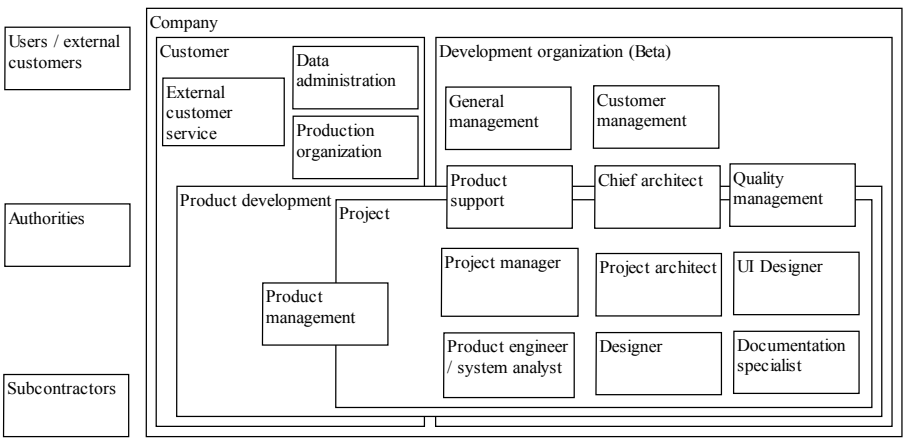

Fig. 2. Map of identified stakeholder roles in Beta

3.1.3 Gamma

The number of Gamma's stakeholder roles (Figure 3) appeared rather low compared to Alfa and Beta. However, after a second glance, the situation gets more compli-

cated. The external stakeholders evidently play the most significant role in Gamma. Customers interact tightly with the development organization and the variation within the set of customers and inside their organizations is evident. Gamma co-operates with customers' data administration organizations, future users, and business representatives who own the delivered systems. In the systems delivery process, Gamma negotiates intensively also with diverging kinds of suppliers: such as other system development organizations interfacing the systems under development, software producers and importers, and hardware vendors.

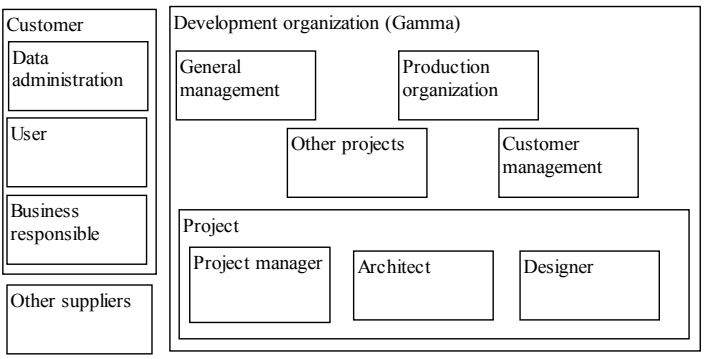

Fig. 3. Map of identified stakeholder roles in Gamma

Gamma's projects seem to be organized quite uncomplicatedly (Figure **3**) compared to Alfa and Beta. This indicates fuzzier role clarity [cf. 24] within projects than in the other two organizations. While the number of explicit roles in architecture design and description process in Gamma seems small, the roles seemed more ambiguous. Especially designers' tasks varied, including even tasks in the customer interface and in architecture design and description. The interviewees were also aware of other development projects within the organization more clearly in Gamma than in Alfa and Beta. Varying technology and diverse customers obliged the projects to change experiences with other projects inside Gamma's large software development organization.

All interviewees in Gamma were quite aware of Gamma's own business and even their customers' business needs. The interaction with customers was intense. The role of customer management thus emerged as a key role in architecture design and description. Architecture must be sold and integrated to the customer's infrastructure and Gamma's customer management and marketing must understand the technological risks emerging during the sales negotiations. The role of general management, in turn, seemed less important. The general management did not deal with technological risks much, having a more business-centric attitude at a general level than in Alfa and Beta.

3.2 Key Stakeholder Roles and Their Rationale for Architecture Description

As it is rather unfeasible to give detailed descriptions of all the stakeholder roles we found (Figure **3**), we focused on identifying and discussing key stakeholder roles and their characteristics in the architecture design and description process. Six roles oc-

curred in all three organizations: 'customer', 'designer', 'project manager', 'architect', 'customer management and marketing', and 'general management'. Hence, we regarded these as the key stakeholders. They were also most frequently mentioned in the data. The role of 'system analyst' was important in Beta as the requirements gatherer. In Alfa, it did not occur at all, and in Gamma it was mixed with the roles of project manager and designer – no separate mentions of 'system analyst' existed in Gamma. Hence, we excluded it from the further analysis. Table 4 lists the key stakeholder roles with their typical tasks and the observed variation within each role. Each stakeholder has a typical relationship and perspective to architecture, which cause certain pertinent problems and challenges. A stakeholder's relationship to architecture and the emerging problems then emphasize certain rationale for each of them (Table 5).

Table 4. The key stakeholder roles in architecture design and description

Stakeholder/role	**Typical tasks**	**Observed variation**
Customer	Buying the system Setting requirements Utilizing the system Running the system	Includes both business and technology oriented people at customer side
Designer	Detailed design Programming	Tasks vary according to skills and experience
Project manager	Project management	Some projects include separate technical and business project managers
Architect	Evaluation of high level solutions Deciding about technology High-level design	Occurrences of more experienced 'chief architects' and less experienced or skilled 'project architects'
Customer management & marketing	Selling Negotiation	Account managers, salesmen
General management	Resource management Deciding strategies	Project steering groups, department managers, team leaders

A customer must comprehend a software system also at a technical and architectural level. The customer builds and operates the technical infrastructure (e.g., data communications and processing services) for the system or purchases it as a service. All this requires communication at an architectural level possibly leading to the problems in technical maturity (e.g. lacking technical skills and knowledge) and in the interpretation of meanings of the technical descriptions. According to numerous mentions in the data, the architecture descriptions were often considered too "technical" for the customer. As well, the problems of trust and security between the customer and the development organization were frequently mentioned. Due to the relationship to architecture and the related problems, communication and understanding were emphasized as the major rationale when customers were mentioned. Naturally, a customer must evaluate the technology and make decisions about it as well as the development organization.

All the organizations used explicitly the term 'designer' instead of 'software engineer', which could also appropriately label this role. Designers create detailed designs

and system implementations according to architecture descriptions. Designers' detailed interest in the parts of the system tends to be somewhat isolated from the whole. This was mentioned as a problem in several occasions. The rationale emphasized in relation to the designer can most clearly be associated with the programming-in-the-large conception. Designers need architecture descriptions for design and implementation, for enabling reuse, and for documentation purposes. It was, however, realized that designers must first be able to understand and interpret the descriptions. Current state-of-the-art in the target organizations showed many imperfections in their description practices thus hindering the interpretation also among the designers.

Project manager was also considered a key stakeholder role in the architecture design and description. Architecture was often mentioned as one of the important sources of information for dividing a project into detailed work assignments. Resource or schedule constraints, such as unrealistic schedules or too little financing, were mentioned as possible causes leading to dubious compromises in architecture. A project manager must use architectural information and decide the critical path and the usage of resources in the subsequent development project. The project manager also discusses and negotiates about the architecture with the other stakeholders.

Table 5. Problems and rationale of the key stakeholder roles

Stakeholder role	Relationship to architecture	Emerging and emphasized problems	Emphasized rationale for architecture description/design
Customer	Varies. Must be able to rationalize the solutions, build and operate the technology infrastructure	Technical maturity Communication and interpretation of meanings and descriptions Security constraints	Communication Understanding Evaluation and deciding
Designer	Implements the software according to architecture	Isolated views Interpretation of descriptions	Understanding Design & implement Reuse Documentation
Project manager	Uses architecture for creating project breakdown structures.	Resource or schedule constraints	Project planning Communication Understanding Evaluation & deciding
Architect	The main creator of architecture	Skills and experience of other stakeholders Lack of resources	Communication Evaluation & deciding Documentation
Customer management & marketing	Uses architecture as a selling argument Demonstrates technical competency and compatibility with architecture	Understanding the descriptions Skills and experience Making resource estimates Isolated views	Communication Understanding Licensing and pricing Resource planning Selling
General management	Must have some understanding and analysis capability concerning technology	Knowledge management Interpretation of descriptions Organizational obstacles	Understanding Communicating Resource planning Quality management

The role called architect was identified as the main creator of architecture in all the three organizations. Architects are invariably technically skilled, being selected among the most technically advanced personnel with a broad view on technology. This causes also problems. No matter how skilled the architects are they must communicate with other stakeholders, who often lack the technological skills and insight, to get their message clear. An architect must be able to communicate with various stakeholders at several levels of abstraction and difficulty with different kinds of descriptions. The evaluation and decision-making on alternative solutions represents another rationale for architecture descriptions by the architects. The observed role of architects corresponded rather directly to Grinter's [14] in-depth observations. In addition, due to the continuous lack of skilled architects, architecture should also be well documented so that the needs for personal presence of the architects the numerous meetings and negotiations of the development process could be minimized.

Customer management and marketing utilizes architecture descriptions in sales negotiations. They must also demonstrate the vendor's technical competency and compatibility to a customer's environment. The employees involved in customer management and marketing may lack technological insight. Hence, the understanding of architecture descriptions may become a problem. Without a realistic picture of architecture, usable resource estimates in customer-vendor negotiations can be hard to reach. The data showed a tendency of cultural and operational isolation between the customer-oriented and technically oriented people. The rationale for architecture design and description among customer management and marketing thus resided in the understanding of architecture at some level, in communicating with the descriptions, in rationalizing the licensing and pricing requirements, in initial planning for development resources, and in "selling" the architecture to customers (in which even the visual attractiveness of the architecture descriptions counts as a factor to be considered).

The general management wanted as well to understand and analyze the technological and architectural issues. The general management had an urge to locate the necessary information concerning the ongoing and planned projects, to encourage the creation of architectural knowledge, and to disperse that knowledge in the organization. The data hinted about possible organizational obstacles, such as barriers between organizational units or different professional backgrounds of the stakeholders that might impede efficient architectural design and description processes. Quality management was considered especially as the rationale held by the general management. The other rationale held by general-level managers corresponded rather closely to those of customer management's.

3.3 Similarities and Differences among the Role Occurrences

Within the three target organizations, the roles of customer management and marketing, designer, and project manager appeared quite similar when observing the rationale for architecture descriptions. Their roles were rather uniformly defined and the emphasized rationale for architecture were well understood among the stakeholders.

More variation in the emphasized rationale for architecture descriptions emerged among architects and customers. An architect's role varied according to questions such as to what extent project management responsibilities were assigned to archi-

tects, or how much an architect was expected to control the conformance to the technological strategies and architectural policies of the organization. In addition, the role of customer included some variation in the rationale. The questions that affected to the variation were, for example: how much architectural descriptions were used during the process of purchasing negotiations, how much the customer's data administration was involved in architecture design, and what was the role of the customer's legacy interfaces and the run-time environment.

The role of general management varied most clearly among the target organizations. In Gamma, general management had only a minor role related to architecture, whereas in Alfa, general management was considered as a key stakeholder related to architecture. An explanation lies in the degree of how much general management was involved in the management and evaluation of technological risks when designing architecture. When the technological and operational risks were decentralized to individual projects (e.g., in Gamma), general management's role related to architecture decreases. On the other hand, when the risks were included in business and technology strategies created by the general management (in Alfa), general management must certainly be regarded as a key stakeholder.

Other stakeholder roles that included considerably variance or that were unique among the organizations were, for example, the roles of data administration or production organization (the importance of run-time architecture), maintenance and support (the role of documentation), other projects (reuse and knowledge management across projects), and other suppliers (communication between vendors, subcontractors, and authorities). The list of observed stakeholders is by far not complete. By extending the sample to additional target organizations, we could evidently find also other unique or varying stakeholder roles not included here.

Table 6. Common rationale for architecture description

Stakeholder role	**Common rationale for architecture description**
Architect	Must communicate about architecture with other stakeholders
Customer	Must communicate about architecture with other stakeholders Must evaluate the architecture and make decisions about it Must use 3rd party products and services associated with architecture
Customer mgmt and marketing	Must communicate about architecture with other stakeholders Must understand the architecture at some level
Designer	Must design and implement the system according to architecture Must understand the architecture
Project manager	Must communicate about architecture with other stakeholders Must plan the project and its resources against architecture

Table 6 lists those rationales that occurred uniformly in the three organizations in relation to a particular stakeholder role. The rationales are quite obviously evidenced by common sense. Nevertheless, they show the importance of communication and the richness of architecture as a concept.

3.4 Implicit Relationships of Architecture, Viewpoints, Stakeholders, and Rationale

Our original objective was to find out the combinations and relationships between rationale, stakeholders, and used viewpoints in the architecture design and description process. We did not define beforehand what we meant by 'architecture'. Based on the initial discussions, we already knew that the concept would carry ambiguous connotations in the organizations and therefore we did not want to constrain the data by artificial definitions. Instead, we let the interviewees to define their own points-of-view to architecture and analyzed the meaning from the transcripts. The concept of architecture carried varying connotations, indeed. To some, it included only the structure of software components, and to some, it included system's stakeholders and their required functionality, packages of business operations, network topologies, among many other features of the system.

With regard to the issue of viewpoints, the interviewees were not explicitly aware of such specific viewpoints recommended or defined in the literature [e.g. 5]. The concept of a viewpoint was not explicitly present in the target organizations' process documents or architecture descriptions, nor recalled in the interviews. On the other hand, so-called "rich descriptions", informally constructed and communicated among the key stakeholders, were used in all kinds of situations; obeying no explicit viewpoint definition [as defined in 16]. These "rich descriptions" were used in negotiations with customers as well as in internal meetings and communication among the stakeholders. Typically, they included PowerPoint™ presentations and extensive textual documents comprising often tens of pages of text and pictures; e.g., imported UML-based models created with design tools such as Rational Rose™, which was used as well for architecture descriptions in all target organizations. Those textual documents were often regarded as "too specific" for other stakeholders than designers.

Furthermore, several interviewees were also rather unaware of the rationales behind the descriptions. Some created architecture descriptions simply because it was defined obligatory in the process model and some were aware only of a proportion of the uses of their descriptions. These reasons led us to abandon the original idea of trying to relate viewpoints explicitly to particular rationale and stakeholders (at least based on our contemporary amount of data).

4 Implications

The observed multitude of stakeholder roles, with their problems and rationale related to software architecture raised a number of implications for practice and research, to be discussed as follows.

4.1 Implications for Practice

The repertoires of stakeholder roles clearly varied in the target organizations. More variation could be probably found by extending the sample. Certain key roles (customer, general management, architect) included more variation in relation to archi-

tecture than others (project manager, designer). Any generic methodology or process model for architecture design thus seems unlikely usable straightforwardly as such in practice. Our findings suggest that organization-specific aspects should be carefully considered in connection to the development of architecture design methods and processes. The process, methods, and tools for architecture design and description should be adapted to the needs of the stakeholders in a particular context. This adaptation could start with an explicit definition for the stakeholders in the organization in question, their architectural tasks, and the rationale they have for architecture design and description. The role of customers and other external stakeholders was highlighted. In our sample, the role of customer affected architecture design and description. The more customer-orientation was observed in architecture design and description, the more emphasis in architecture descriptions was put on communicational aspects and interpretation, instead of the detailed design and implementation.

Architecture design and description should not be approached separately from the other processes of software engineering, representing the core of work for a part of the key stakeholders. These processes include at least requirements management, project management, customer management, and technology management. In the target organizations, architecture emerged as a rich concept with a multitude of meanings for diverging stakeholders, and its design and description was intertwined with all these other processes, which provided additional and necessary knowledge and understanding for architecture design, and vice versa. The plain viewpoint of the 'architect' thus cannot explain the architecture design process in itself.

We noticed that designers, together with plainly technologically oriented architects, could be satisfied with current architecture description practices, although the descriptions were varying and their contents were all but formalized or standardized. Other stakeholders, such as management, were more concerned with standardized practices. The management worried clearly more about the reuse and the aspects of knowledge creation and management, whereas the technical people were relatively happy with the current situation, project-by-project. There are several possible explanations to this phenomenon. Managers have a wider view to architecture, including also the business and resource management aspects, being unable to locate that information in relation to the contemporary descriptions. They also needed to coordinate a number of projects. Moreover, architecture descriptions were traditionally designed to serve designers, rather than managers. The detailed architecture descriptions for the implementation purposes were too specific for managers, hindering quick overviews of the general situation across projects. The information needed from the architecture descriptions thus greatly varied depending on the diverging use situations, e.g., if the aim is to communicate and make business decisions versus the information needed for designing and implementing the system (i.e., programming-in-the-large). The stakeholders of the architecture design and description process must be aware of these differences between the needs and conceptions of the stakeholders in order to achieve common understanding of the varying rationale for architecture design and description.

4.2 Implications for Research

Our study suggests that further research on the software architecture design and description should consider a wider scale of issues than the programming-in-the-large rationale or the plain viewpoint of the software architect. Current conceptions on architecture design and description, including the proliferation of the formal languages and models that currently dominates the research on software architecture are somewhat based on such presuppositions about the rationale for architectural design that do not correspond with practice.

The rationales most emphasized among the key stakeholders were related to the issues of communicating and understanding the architecture, and making evaluations and decisions about it. The practical problems mentioned were rather unassociated with achieving the "right" or "most efficient" solution. Instead, they were mostly associated with communication, interpretation of the descriptions, skills and experience, and other "soft" aspects of software engineering. The aim to design and implement the system according to an architecture description appeared to be central only to designers. An architecture description was primarily considered a means for creating common understanding about the technological solutions and about the way of structuring associated with the system. The multiple stakeholders with multiple rationales identified in the target organizations call for empirical research on communication in architecture design and how it could be enhanced. The research should cover the evaluation of the role of formal and informal models and descriptions in architecture design – to what extent knowledge is created and transferred through formal models and to what extent through informal pictures, textual descriptions, and other social interaction. After that, we can determine effective means and tools to develop the architecture design and description process. These means might slightly differ from the current focus on formal issues of architecture design.

Our study thus sets modest challenges to the contemporary work around ADLs. If the emergence of informal "rich descriptions" of architecture aiming mostly at facilitating communication between diverse stakeholders increases among the software-producing organizations, the ADL developers might want to elaborate visual, flexible, communicative, and collaborative tools that could support all stakeholders (alongside architects and designers), still trying to retain the advantages of the programming-in-the-large metaphor and formality that have been already reached. On the other hand, UML was used extensively in all the three organizations, and designers and some of the architects seemed quite content with UML's ability to model architecture. This raises a paradox: if designers and architects remain unaware of the advantages of specific formal ADLs, then who would or should be aware of them, and what would be the practical rationale for those in the first place? As long as the lack of reported experiences from applying ADLs in real-life software producing companies continues, their contribution remains at a somewhat theoretical level. Practice-oriented research strategies, e.g. that of action research [see e.g. 25], might fruitfully shed additional light on this issue.

5 Conclusions

Different stakeholders have different purposes for architecture and they emphasize different rationales for architecture description. Whereas only designers emphasized architecture as a basis for further design and implementation, the other stakeholders emphasized it rather as a means for communication, interpretation, and decision-making. This observation challenges the common conception of architecture design as programming-in-the-large. Instead, architecture design and description can be regarded primarily as a means for coping with complex solutions and technology, reached through communication between diverse stakeholders with varying skills and experience. The primary focus of the software architecture research should be shifted onto the non-technical rationale and common understanding of technical complexity of software in relation to the relevant organizational and business environments.

References

1. Medvidovic, N. and Taylor, R. N., "A Classification and Comparison Framework for Software Architecture Description Languages," *IEEE Transactions on Software Engineering*, vol. 26, no. 1, 2000, pp. 70-93.
2. Luckham, D. C. and Vera, J., "An Event-Based Architecture Definition Language," *IEEE Transactions on Software Engineering*, vol. 21, no. 9, 1995, pp. 717-734.
3. Allen, R. and Garlan, D., "A formal basis for architectural connection," *ACM Transactions on Software Engineering and Methodology*, vol. 6, no. 3, 1997, pp. 213-49.
4. DeRemer, F. and Kron, H. H., "Programming-in-the-Large Versus Programming-in-the-Small," *IEEE Transactions on Software Engineering*, vol. SE-2, no. 2, 1976, pp. 80-86.
5. Kruchten, P. B., "The 4+1 View Model of Architecture," *IEEE Software*, vol. 12, no. 6, 1995, pp. 42-50.
6. Monroe, R. T., Kompanek, A., Melton, R., and Garlan, D., "Architectural styles, design patterns, and objects," *IEEE Software*, vol. 14, no. 1, 1997, pp. 43-52.
7. Hofmeister, C., Nord, R., and Soni, D., *Applied Software Architecture*. Reading, MA: Addison-Wesley, 1999.
8. Gomaa, H., Menascé, D. A., and Shin, M. E., "Reusable Component Interconnection Patterns for Distributed Software Architectures," *ACM SIGSOFT Software Engineering Notes*, vol. 26, no. 3, 2001, pp. 69-77.
9. Kazman, R., Abowd, G., Bass, L., and Clements, P., "Scenario-Based Analysis of Software Architecture," *IEEE Software*, vol. 13, no. 6, 1996, pp. 47-55.
10. Kazman, R., Barbacci, M., Klein, M., Carrière, S. J., and Woods, S. G., "Experience with Performing Architecture Tradeoff Analysis," Proceedings of the 1999 International Conference on Software Engineering, 1999, pp. 54-63.
11. Bosch, J., Design and Use of Software Architectures: Adopting and Evolving a Product-Line Approach: Addison-Wesley, 2000.

12. Robbins, J. E. and Redmiles, D. F., "Software Architecture Critics in the Argo Design Environment," *Knowledge-Based Systems*, vol. 11, no. 1, 1998, pp. 47-60.
13. Schön, D., The Reflective Practitioner: How Professionals Think in Action. New York: Basic Books, 1983.
14. Grinter, R. E., "Systems Architecture: Product Designing and Social Engineering," *ACM SIGSOFT Software Engineering Notes*, vol. 24, no. 2, 1999, pp. 11-18.
15. Shaw, M., "The Coming-of-Age of Software Architecture Research," Proceedings of the 23rd International Conference on Software Engineering (ICSE 2001), 2001, pp. 657-664a.
16. IEEE, "IEEE Recommended Practice for Architectural Description of Software-Intensive Systems," IEEE, IEEE Std 1471-2000, 2000.
17. Garlan, D., "Software Architecture: a Roadmap," in *The Future of Software Engineering*, A. Finkelstein, Ed.: ACM Press, 2000.
18. Bass, L., Clements, P., and Kazman, R., *Software Architecture in Practice*: Addison-Wesley, 1998.
19. Strauss, A. L. and Corbin, J., *Basics of Qualitative Research: Grounded Theory Procedures and Applications*. Newbury Park, CA: Sage Publications, 1990.
20. Galliers, R., "Information Systems Research: Issues, Methods and Practical Guidelines." Oxford: Blackwell, 1992.
21. Nandhakumar, J. and Jones, M., "Too Close for Comfort? Distance and Engagement in Interpretive Information Systems Research," *Information Systems Journal*, vol. 7, no., 1997, pp. 109-131.
22. Glaser, B. and Strauss, A. L., The Discovery of Grounded Theory: Strategies for Qualitative Research. Chigago: Aldine, 1967.
23. Miles, M. B. and Huberman, A. M., *Qualitative Data Analysis: A Sourcebook of New Methods*. Beverly Hills: Sage, 1984.
24. Goodman, R. A. and Goodman, L. P., "Some Management Issues in Temporary Systems: A Study of Professional Development and Manpower - The Theater Case," *Administrative Science Quarterly*, vol. 21, no. Sep 1976, 1976, pp. 494-501.
25. Avison, D., Lau, F., Myers, M. D., and Nielsen, P. A., "Action Research," *Communications of the ACM*, vol. 42, no. 1, 1999, pp. 94-97.