

Using Distributed Cognition Theory for Analyzing the Deployment Architecture Process

Naomi Unkelos-Shpigel and Irit Hadar

Information Systems Department, University of Haifa
Carmel Mountain 31905, Haifa, Israel
{naomiu, hadari}@is.haifa.ac.il

Abstract. Deployment architecture is an important part of the software development lifecycle. Our preliminary research indicates that the process of constructing the deployment architecture holds several challenges which, if not properly met, may hinder the success of the project and result in low customer satisfaction. In this ongoing research, we analyze this process in different firms, in an attempt to understand its challenges towards proposing strategies for its improvement. To this end, we collected data via 25 questionnaires at a global IT firm, and 12 in-depth interviews with architects from 10 firms. Analyzing the data qualitatively, through the lens of distributed cognition theory, helped us to understand the structure and flow of this multiple-stakeholders process and identify common potential pain points and challenges that need to be addressed.

Keywords: deployment architecture, distributed cognition, qualitative research.

1 Introduction

Deployment architecture is an important part of the software development lifecycle. Its purpose is to bridge the gap between the requirements of the capabilities and the solution delivered and installed at the customer's site [1]. It is defined as "allocation of the system's software components (and connectors) to its hardware hosts" [2, p.1], and highly influences the quality of service (QoS) [2].

Since the deployment solution involves adjusting the product to the customer's needs and environment, it has a major effect on customer satisfaction of the product [3]. Several empirical studies have examined the deployment process, mostly focusing on creating the deployment architecture and understanding its requirements [4]. Deployment architecture concerns that are not widely inquired include the challenges along different phases of the deployment process and the different tools and support the service architect (who creates the architecture, hereafter referred to as the architect) can rely upon. These concerns may influence the decisions made by the architect while constructing the deployment architecture solution and specifically, the components selected to be part of this solution.

The objective of this ongoing research is to enhance the deployment architecture process, resulting in better solutions and higher customer satisfaction. Accordingly, as a first step, our empirical study aims at identifying the challenges and difficulties that

may hinder its success. To this end, we study the flow of the architecture deployment process, its different participants, and their decision-making and reasoning processes.

Some attention has been given in the research areas of software and IS engineering to cognitive processes related to different activities of development. For example, a survey on cognitive studies in the context of software analysis and design can be found in [6]. However, we could not find empirical studies investigating cognitive processes of software architects in general and of deployment architects in particular. In this study, we make a first step towards investigating the construction of deployment architecture from a cognitive point of view. Considering that the creation of deployment architecture involves several different sources – both human and non-human – we recruited for our analysis the theory of distributed cognition.

Distributed cognition separates a system, e.g., a collection of people and tools working together to achieve a goal, to subsystems, referring to the cognition of each subsystem in the process as a separated unit [5]. It addresses the concerns of the different subsystems, their interaction, conflicts and dilemmas that stem from their different points of views. Distributed cognition further proposes strategies to improve the systems and its processes based on the identified conflicts. This theory fits very well to our objective since the creation of the deployment architecture solution is a complex process, which involves different stakeholders and various tools and artifacts. Addressing each of these as a unique subsystem, with its own cognition and perception of the process, can provide us a better understanding of the process, its weaknesses, and accordingly, opportunities for its improvement.

2 Distributed Cognition

Distributive cognition is based on the principle that "cognitive processes are socially distributed across the members of a group" [7]. It extends the reach of what is considered cognitive beyond the individual, and relates to the interactions between people, materials and resources. The theory looks for a broad class of cognitive events and does not expect all such events to occur within the physical boundary of an individual, in contrast with other popular cognitive theories [7]. Social organizations are considered forms of cognitive architecture. A system can dynamically configure itself, bringing subsystems into coordination to accomplish various goals. A cognitive process is delimited by the functional relationships among the elements that participate in it, rather than by the spatial collocation of the elements. The context of an activity determines the way information flows through a group. Memory is distributed among various subsystems, and each unit (subsystem) has its own cognition, according to the actions it performs; this can lead to some contradictions in the perceptions of different units [8]. The study of cognition is not separable from the study of culture; culture provides us with intellectual tools that enable accomplishing things that could otherwise not have been completed [7].

The study of distributed cognition is usually done ethnographically, focusing on the meaning of the systems and the meaning of the words used during the process [7]. In contrast to classic cognitive theories, the focus here is not only on the knowledge

of people, but also on how they use what they know in order to do what they do. The theory claims that cognitive activity is constructed from both internal and external resources and that the meanings of actions are grounded in the context of an activity. This theory can be used to improve the design of working processes [8].

In the domain of IS engineering, this theory has been used mainly in HCI research [7,9]. Nillson et al. [9] used distributed cognition to analyze human-machine interaction, and the term socio-technical system, which includes the people carrying out a task and the artifacts they use. This created a new perspective on the structure and role of the subsystem in the process; a subsystem can include both human and non-human parts, and interact with other such subsystems [9].

3 Empirical Study

3.1 Method and Settings

The main objective of our study is to identify the challenges and difficulties that may hinder the quality of the deployment architecture solution. To this end, we conduct a qualitative research, where the investigator does not have a pre-defined theory about the environment inspected, but rather uses techniques and tools to explore and discover phenomena [10]. We use tools and methods based on the grounded theory methodology [10]; data is collected from the field and inductively analyzed.

We initiated our research at a global, large-scale IT firm by a pilot of interviews, for a first acquaintance with the firm and its deployment division, followed by distributing an open-questions questionnaire to deployment architects. The questionnaire was aimed to achieve a preliminary identification of the major concerns of the process and opportunities for improvements, as perceived by architects. In order to enrich our data beyond this single case study [11] we collected data via in-depth interviews in additional firms. All together, ten global firms – six hi-tech firms, two communication firms and two shipping firms – were included in the research so far. Only global firms that regularly execute major deployment projects were included in the research. The interviewees were experienced deployment architects, working with customers as well as with internal divisions responsible for defining products. In what follows we present preliminary results obtained thus far in this research.

3.2 Analysis and Findings

Using an inductive qualitative data analysis method [10], we divided the text from the interviews transcriptions and questionnaires to statements and classified them to categories of challenges within the process. All together, 80 relevant statements were identified and classified. Next, we mapped the emergent categories to the unit of analysis as defined by the distributed cognition theory. When applying grounded theory, consideration of literature is allowed for guiding data analysis [12].

The first and most profound principle of distributed cognition is to determine the unit of analysis, in which cognition will be examined. Since we explore the process of deployment architecture, we defined the unit of analysis to include all stakeholders of this process, as well as the resources and systems used. The different subsystems included in our unit of analysis, based on the data collected, are described in figure 1.

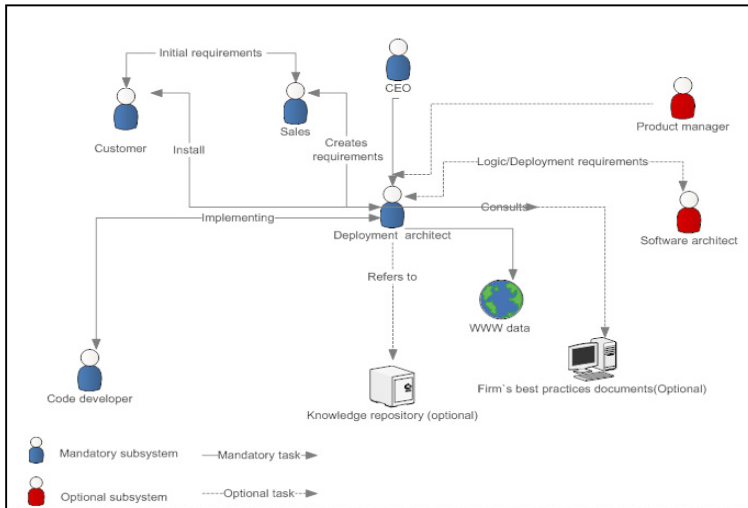


Fig. 1. Subsystems in the deployment architecture process

In what follows we present and briefly explain each category and its related challenges. Due to space limitations, we will not present here exemplary quotes.

Memory Distributed among Subsystems: Having the memory distributed among subsystems, as explained by the distributed cognition theory, results in different subsystems having different view of the process [7]. This often leads to conflicts and contradictions between subsystems.

Challenge: Having organizational memory distributed among different subsystems causes conflicts during the process. It leads each subsystem to focus on its own interests and priorities. For example, architects often find themselves constrained to promises given by salespersons, which in some cases are not provided by the product. This happens usually because the communication between these two subsystems occurs at a late stage, after all the requirements of the product have been established. Such cases result in either high costs of patching solutions, or in failure to deliver.

This challenge is typical to firms not using a knowledge repository.

Internal and External Representations of Concepts/Artifacts: Each subsystem has its individual cognitive processes based on its own knowledge and point of view, in addition to the information communicated by other subsystems [7]. This results in interpretation of an external representation of information (e.g., requirements document) to an internal, often different representation of this information, which affects the action taken by the subsystem.

Challenge: The difference between internal and external representations often causes misunderstandings during the process. For example, when different stakeholders refer to the same item using different terminology. As a result, major issues regarding product features may be misinterpreted and thus not delivered as intended. This challenge was found in all the firms in our study except for one. In this firm, a separate team was responsible for creating an overall taxonomy for the process; all stakeholders were instructed to use these unified terms.

Interaction among Subsystems: The different subsystems communicate with each other in order to complete the process successfully [5,p.251]. However, this dependency on communication might lead to tension and conflicts between stakeholders.

Challenge: We encountered in several cases that the communication between code developers and project management is weak or non-existent. Thus, managers are often not informed about problems related to product development. Only later, during the deployment, the product is found to be not eligible to be deployed at customer's site: Another example of missing communication between subsystems is related to lack of information about the quality of different components of the product, which failed tests in the context of other stakeholders. This problem was mentioned by many of the architects; we found no solution for this problem in any of the firm we studied.

Interface with Tools: Interaction between human and non-human subsystems and the effect of the latter on the cognitive processes of the former is another focus of distributed cognition. As illustrated in figure 1, the deployment architect uses other subsystems, which are automated tools such as remotely accessible knowledge repositories or the Internet. This eliminates the need in spatial proximity between subsystems, and implies that the context of action (the purpose of using the tool) denotes its meaning [7].

Challenge: Due to the differences in focus, different stakeholders use different tools to create and manage information. While the architect should be able to see all different views, no single tool can display all the views of all the stakeholders at the same time. This causes the architect to waste additional time on looking for data in multiple tools and systems, viewing only fragments of the information at a time, which makes it difficult to see the bigger picture. Moreover, some tools are over complex for their purpose, leading stakeholders to work with simpler tools (e.g., excel), which are sufficient for them but do not provide the wide perspective needed for the architect. This challenge is typical to firms, which do not have a holistic knowledge management solution, with a knowledge repository and appropriate tools for each stakeholder.

Cultural Environment: Culture influences cognitive processes both within and between subsystems [7].

Challenge: A good example, which we witnessed in several American firms, although global in nature, is the tendency to neglect addressing international regulations. Firms failing to consider these factors, usually when management does not put enough emphasis on the matter and due to the natural focus on the local market, often face severe problems when deploying their product in countries where unique regulations on this type of product are defined. This challenge was mostly found in firms based in countries with rigid regulations. Firms addressing this concern as part of their working process, instruct the participants of the deployment process to address international regulations at an early stage, and construct the solution accordingly.

4 Conclusion

This paper demonstrated the analysis of the deployment architecture process through the lens of distributed cognition. Using this theory we were able to refer to each of the

stakeholders and tools as a separate subsystem, having its own cognition and interactions with other subsystems. We used this perspective to analyze the data we collected on the deployment architecture process, and to identify the challenges deployment architects face and their sources.

In contrast to typical empirical studies on distributed cognition, based mostly on observations, we based our findings on statements taken from interviews and questionnaires. However, since we used two different data collection tools – questionnaires and interviews, and different data sources spreading over 10 firms from different domains, we believe this limitation is mitigated, at least to some extent, by this triangulation. Having said that, it should be noted that this is still an ongoing study, and that the findings presented here are merely preliminary. We plan to refine and validate our findings via additional interviews and observations. Next, we will develop strategies for improving the deployment architecture process based on our findings and on the principles of the distributed cognition theory.

References

- [1] Carzaniga, A., Fuggetta, A., Hall, R.S., Heimbigner, D., Van Der Hoek, A., Wolf, A.L.: A characterization framework for software deployment technologies. Technical Report. Colorado State Univ. For Collins Dept. of Computer Science (1998)
- [2] Medvidovic, N., Malek, S.: Software deployment architecture and quality-of-service in pervasive environments. In: International Workshop on Engineering of Software Services for Pervasive Environments, in Conj. with the 6th ESEC/FSE Joint Meeting. ACM (2007)
- [3] Mockus A., Zhang P. and Li L.: Predictors of customer perceived software quality. In: Proceedings of the International Conference on Software Engineering (ICSE 2005), pp. 225–233. ACM (2005)
- [4] Adam, S., Doerr, J.: On the notion of determining system adequacy by analyzing the traceability of quality. In: Proceedings of Workshops and Doctoral Consortium (2007)
- [5] Hutchins, E.: *Cognition in the Wild*. MIT Press, Cambridge (1995)
- [6] Hadar, I.: When Intuition and Logic Clash: The Case of the Object Oriented Paradigm. *Science of Computer Programming* (in press)
<http://dx.doi.org/10.1016/j.scico.2012.10.006>
- [7] Hollan, J., Hutchins, E., Kirsh, D.: Distributed cognition: toward a new foundation for human-computer interaction research. *ACM Transactions on Computer-Human Interaction (TOCHI)* 7(2), 174–196 (2000)
- [8] Ackerman, M.S., Halverson, C.: Considering an organization’s memory. In: Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work, pp. 39–48. ACM (1998)
- [9] Nilsson, M., Laere, J.V., Susi, T., Ziemke, T.: Information fusion in practice: A distributed cognition perspective on the active role of users. *Information Fusion* 13(1), 60–78 (2012)
- [10] Strauss, A., Corbin, J.: *Basics of qualitative research grounded theory procedures and techniques*. Sage Publications, Inc. (1990)
- [11] Walsham, G.: Interpretive Case Studies in IsResearch: Nature and Method. *European Journal of Information Systems* 4(2), 74–81 (1995a)
- [12] Suddaby, R.: From the editors: What grounded theory is not. *Academy of Management Journal* 49(4), 633–642 (2006)