When a Paradigm is Inconsistent with Intuition: The Case of Inconsistency Management

Irit Hadar and Anna Zamansky[(✉)]

Information Systems Department, University of Haifa,
A. Hushi 199, 3498838 Haifa, Israel
{hadari,annazam}@is.haifa.ac.il

Abstract. The acceptance and correct use of new paradigms in information systems engineering is highly affected by cognitive dispositions of the individual engineers. In particular, the engineers' intuition – based on their vast experience – may come in the way of accepting the rationale of a new paradigm. Analyzing engineers' reaction to a newly introduced paradigm through the lens of cognitive psychology may highlight potential barriers to the paradigms' successful adoption and use. In this position paper we demonstrate the potential benefit of the above-proposed approach via a case study focusing on the paradigm of inconsistency management. We present our findings, based on in-depth interviews with 20 practitioners, and analyze them using the dual-process theory, highlighting a clear clash between their intuition and the paradigm's rationale.

Keywords: Cognitive processes · Paradigm acceptance · Dual-process theory · Inconsistency management

1 Introduction

New paradigms for information systems engineering are constantly introduced. Organizational processes and related aspects, such as organizational climate and culture, play an important role in new paradigms' acceptance and adoption, and are vastly discussed in literature. However, a necessary condition for a successful adoption is the individual engineers' acceptance of the paradigm and their ability to correctly use it, which highly depends on their cognitive disposition. This aspect of paradigm acceptance has received far less attention thus far. Analyzing paradigms through the lens of cognitive psychology may provide new insights into the cognitive aspects that may hinder the paradigms' successful adoption and use. In order to demonstrate the potential benefit of this approach, in this position paper we discuss, as an example, the paradigm of inconsistency management, which has been gaining attention in recent years in the field of information systems.

Handling inconsistencies is a key challenge in the process of information systems (IS) development. Inconsistency may occur, for example, when requirements or specifications contain conflicting or contradictory descriptions of the expected behavior

of the system or of its domain. Such conflicting descriptions may come, for example, as a result of conflicting goals between different stakeholders, changes introduced during the evolution of the requirements, etc. [11].

Until recently, inconsistencies in IS development were viewed as a problem that needs to be eliminated before further activities can take place. Over the last two decades, however, a more tolerant approach toward inconsistency has evolved (e.g., [2, 3, 11, 13]. In "Leveraging inconsistency in software development" Nuseibeh et al. [11] argue: "Maintaining consistency at all times is counterproductive. In many cases, it may be desirable to tolerate or even encourage inconsistency to facilitate distributed teamwork and prevent premature commitment to design decisions" (p. 24). Finkelstein [3] further advocates this paradigm shift, proposing that rather than removing inconsistency, we need to manage it by "preserving inconsistency where it is desirable to do so, identifying inconsistency at the point where decisions are required and removing (or otherwise remedying) inconsistency prior to taking action. This requires a major change in the way we think" (p. 2).

But changing the way we think is not always simple. In our ongoing research on practitioners' perceptions regarding a paradigm of inconsistency management, we found that practitioners strongly reject the very idea of not immediately eliminating inconsistency, many times even when they admit that it makes sense. Cognitive psychology theories provide a possible explanation for this phenomenon: Our thinking processes rely many times on automatic responses based on our vast past experience, roughly corresponding to what is commonly called intuition, rather than on analytical reasoning [9, 14]. This intuition may hinder integrating new ideas, as reasonable as they may be, into our thinking processes. Therefore, an in-depth understanding of this cognitive phenomenon is a necessary step toward 'changing the way we think.' In this paper we demonstrate this cognitive analysis on the case of inconsistency management based on empirical evidence, reflect on its implications, and propose directions for future research.

2 Preliminary Observations

2.1 Data Collection

The original objective of the empirical study was to understand practitioners' perceptions regarding the paradigm of inconsistency management. Due to the exploratory nature of our study, we took a qualitative research approach [10]. More specifically, we used the grounded theory methodology, which is appropriate in order to generate descriptive or explanatory theory [15]. When applying grounded theory, consideration of literature is allowed for guiding data analysis [16].

Twenty participants were carefully selected and interviewed in our study. Participants' sampling was performed according to the theoretical sampling principles [15]. The sampling criteria we defined were as follows: Only participants with academic background in IS or computer science, and professional experience in industry and/or in research in IS, were selected. In order to reflect variations within our data, we also aimed to achieve a diverse sample of participants, with different levels of experience

and from different domains. The participants' sample included software developers of different seniorities and different roles, including: software architects, requirements analysts and project managers. The participants' professional experience was 13 years in average, varying from three to 35 years.

The main tool of data collection was semi-structured in-depth interviews, which were conducted by the authors. The interview questions focused on the notion of inconsistency in the context of IS. We asked the interviewees to explain what inconsistency is and how it is manifested in IS, and scrutinized their perceptions and attitudes toward its different manifestations. The full result set obtained in this study is beyond the scope of this paper. Here we focus on the data related to the strong rejection of the ideas of inconsistency management demonstrated by the study participants, and its interplay with their own analytical reasoning about this topic.

2.2   Cognitive Analysis of the Empirical Results

Already during the first interviews, we found that participants react very strongly toward inconsistency, and specifically could not accept the possibility of tolerating it. When asked how inconsistency should be handled, in the context of examples brought up by the participants, their answers were typically uncompromising. Examples of answers we received from different participants include: "*We must eliminate [the inconsistency] on sight.*"; "*[The inconsistency] must always be resolved. ASAP.*"; "*It [the inconsistency] must be fixed. It must be clean.*"; "*As soon as we find it [the inconsistency] it needs to be investigated and solved.*"

Witnessing this strong uncompromised reaction, we decided to present to the participants a case in which it makes sense to tolerate inconsistency, based on the inconsistency management literature [11]: "The observation that some inconsistencies never get fixed seems counterintuitive at first. […] Many local factors affect how you handle an inconsistency, including the cost of resolution, the cost of updating the documentation, and the developers' level of shared understanding. Ultimately, the decision to repair an inconsistency is risk-based. If the cost of fixing it outweighs the risk of ignoring it, then it makes no sense to fix it" (p. 27). Accordingly, we asked the following question: Imagine there is a case in which there is an inconsistency in requirements relevant only to rare cases, however resolving this situation would be of high cost. What would you do?

Following this question, we observed different reactions; while answering, the participants expressed their thoughts out loud, reflecting the thinking process that took place. We observed two patterns of response:

1. Immediate acceptance of the rationale of tolerating inconsistency, despite having rejected it until that point of the interview. For example:

- "*The fact that there are two contradictory requirements doesn't mean that they are important. Perhaps they refer to situations most users won't encounter, and then perhaps they do not even have to be fixed.*"(Developer, three years of experience)
- "*I guess that in such situations one can consider leaving it [the inconsistency] as is, and of course document [its existence].*" Notably, a far less tolerant attitude toward

inconsistency was reflected in this participant's' answers to previous questions, e.g.: *"One has to think ahead and not get into these situations [of inconsistency]. But, it's not always possible, therefore it has to be resolved on sight."* (Developer, six years of experience)

– *"In this case, the inconsistency may be tolerated. If it has no legal meaning, it's possible to leave it."* Here, too, a less tolerant attitude toward inconsistency was reflected previously, e.g.: *"Immediately when it [an inconsistency] is detected, the problem must be investigated; its source must be found and resolved."* (Project manager, 17 years of experience)

2. Transitioning from strong rejection to reluctant acceptance.

– *"No one would ever agree to live with it [the inconsistency]. I, for sure, wouldn't be able to live with it. [Pause] It's all about cost verses benefit. It it's a minor problem, it would ignored."* (Senior developer, 10 years of experience)
– *"I can't see how that's possible. In my case, there are no situations like this. If it's a primary issue, I don't see how it can be ignored regardless the cost. If it's something small, it may be possible. The final answer depends on the type of inconsistency."* (Senior architect, 16 years of experience)
– *"There is no such thing! It shows a severe failure. [Pause] It's all about matching expectations. It's about the contract between you and the customer."* (Chief architect, 20 years of experience).

In both patterns we see that the participants changed their minds from their previous general disposition of intolerance toward inconsistency to a more tolerant attitude. Moreover, in the second pattern we observe a significant change of view during the participants' thinking processes, without any external interaction. What are the cognitive mechanisms that may account for this change? One of the mainstream theories in cognitive psychology that may explain this phenomenon is the *dual- process theory* [9, 14].[1] According to this theory, our cognition operates in two different modes, called System 1 (S1) and System 2 (S2), roughly corresponding to the notions of intuitive and analytical thinking respectively. S1 processes are characterized as being fast, automatic, effortless, unconscious, and inflexible (difficult to change or overcome). In contrast, S2 processes are slow, conscious, effortful and relatively flexible, and serves as monitor and critic of the fast automatic responses of S1. There are situations in which S1 produces quick automatic non-normative responses, while S2 may or may not intervene in its role as monitor and critic [9].

Analyzing the data through the lens of the dual-process theory, we can explain the change in the participants' attitude toward the presented case of inconsistency tolerance as the intervention of S2, namely the analytic reasoning, in the thinking process, overriding the initial S1 response made based on the participants' intuition that inconsistency is an evil to be eliminated on sight.

The data further indicate that while the participants typically understood the rationale of tolerating inconsistency in certain cases and agreed with it, employing S2,

---

[1] There are many theories about the dual process of the human mind. We rely here on the research program of 30 years by Kahneman and Tversky, summarized in [9].

in terms of the dual-process theory, they struggled to reconcile the conflict between the two systems. This struggle was reflected, for example, in the following quotes:

– *"If I have a way to confirm, that this [the inconsistency] is indeed very esoteric, and that it would indeed cost a lot to fix it, then it may be tolerated. It's a decision of the higher management. But it seems to me very individual. I would for sure fix it. But it can be a strategic decision [to tolerate the inconsistency]. Ideally, clearly it should be fixed. In reality there are additional considerations, mainly economic ones."*
– *"It [the inconsistency] can be tolerated. I, personally, would fix it in any case, because things need to be consistent. Consistency is important for software."*

This tension between S1 and S2 is a known phenomenon, as vividly described by Gould [5], referring the normative decision in the famous Linda problem[2]: "yet a little homunculus in my head continues to jump up and down, shouting at me [that the normative decision cannot be correct]" (p. 469). Our participants explain that although they understand the rationale of tolerating inconsistency, they would not have felt comfortable following it.

3   Discussion and Future Research

'Changing the way we think', as suggested by [8], is not always simple. It is not enough to rationally accept an idea that contradicts one's intuition; people consistently make non-normative decisions that contradict knowledge they evidently hold [9]. This was also found specifically in the field of software development [6, 7]. Moreover, the stronger S1 is, the more difficult it is for S2 to override it. In our case, S1 was based on the participants' vast experience of software development, which led them to believe that "*Consistency is important for software*" to an extent that does not allow compromise under any circumstances. This is usually not a bad heuristic, and the reason that this heuristic exists lies in the vast experience that formulated it. For this reason, S1 decision-making is generally more often than not the correct one [4, 17]. However, there are cases, such as the one we witness here, where a heuristic that is usually useful, needs to be set aside.

Our impression was that the participants' years of experience and their role might affect their pattern of behavior. More specifically, we saw that developers with many years of experience were more reluctant to tolerate inconsistency, while product managers and less experienced developers found it easier to accept this rationale. A quantitative research is required to confirm these hypotheses. Based on the explanation above such a finding would make sense; the more experience participants have in software development, the stronger their S1 influence is in this matter.

Our findings highlight the importance of exploring practitioners' intuition when introducing a new paradigm. Importantly, the identification of potentially counter-intuitive principles of a paradigm is only the first step, the next being exploring means for reconciling these principles (as understood by S2) with intuition (S1). Therefore,

---

[2] A frequently cited example in this field: http://en.wikipedia.org/wiki/Conjunction_fallacy.

another important direction for future research is exploring means for overcoming the conflict between S1 and S2, toward an acceptance of new paradigms that may cause this conflict. An approach that could be beneficial to this end is the one proposed in the context of mathematical education: To help students bridge the gap between S1 and S2 thinking, one needs to bridge down the analytical solution to the students' intuition by devising and presenting a bridging task which would be logically (roughly) equivalent to the original analytical task, but psychologically much easier to accept (i.e. closer to intuition)   [1]. In the information systems field, constructing bridging means for overcoming a paradigm's conflict with intuition may go a long way toward wider acceptance of the paradigm in practice.

It is our hope that this position paper will initiate a discourse on the cognitive concerns that are involved in new paradigms' adoption. We intend to continue the research presented in this paper, for further validation and generalization of its results in the context of inconsistency management.

References

1. Ejersbo, R.L., Leron, U., Arcavi, A.: Bridging intuitive and analytical thinking: four looks at the 2-glass puzzle. Learn. Math. **34**(5), 2–7 (2014)
2. Ernst, N.A., Borgida, A., Mylopoulos, J., Jureta, I.J.: Agile requirements evolution via paraconsistent reasoning. In: Ralyté, J., Franch, X., Brinkkemper, S., Wrycza, S. (eds.) CAiSE 2012. LNCS, vol. 7328, pp. 382–397. Springer, Heidelberg (2012)
3. Finkelstein, A.: A foolish consistency: technical challenges in consistency management. In: Ibrahim, M., Küng, J., Revell, N. (eds.) DEXA 2000. LNCS, vol. 1873, pp. 1–5. Springer, Heidelberg (2000)
4. Gigerenzer, G., Todd, P.M.: The ABC Research Group: Simple Heuristics that Make us Smart. Oxford University Press, New York (1990)
5. Gould, S.J.: Bully for Brontosaurus: Reflections in Natural History. Norton, New York (1992)
6. Hadar, I., Leron, U.: How intuitive is object oriented design? Commun. ACM **51**(5), 41–46 (2008)
7. Hadar, I.: When intuition and logic clash: the case of the object oriented paradigm. Sci. Comput. Program. **78**, 1407–1426 (2013)
8. Finkelstein, A.C., Gabbay, D., Hunter, A., Kramer, J., Nuseibeh, B.: Inconsistency handling in multiperspective specifications. IEEE Trans. Software Eng. 208569578 (1994)
9. Kahneman, D. (Nobel Prize Lecture): Maps of bounded rationality: a perspective on intuitive judgment and choice. In: Les Prix Nobel, Frangsmyr, T. (eds.) 416–499 (2002). www.nobel.se/economics/laureates/2002/kahnemann-lecture.pdf
10. Myers, M.D., Avison, D.: Qualitative research in information systems. Manag. Inf. Syst. Q. **21**, 241–242 (1997)
11. Nuseibeh, B., Easterbrook, S., Russo, A.: Leveraging inconsistency in software development. Computer **33**(4), 24–29 (2000)
12. Nuseibeh, B., Easterbrook, S., Russo, A.: Making inconsistency respectable in software development. J. Syst. Softw. **58**(2), 171–180 (2001)
13. Spanoudakis, G., Zisman, A.: Inconsistency management in software engineering Survey and open research issues. In: Chang, S.K. (ed.) Handbook of Software Engineering and Knowledge Engineering, pp. 329–380. World Scientific Publishing Co., Singapore (2001)

14. Stanovich, K.E., West, R.F.: Individual differences in reasoning: implications for the rationality debate. Behav. Brain Sci. **23**, 645–726 (2000)
15. Strauss, A., Corbin, J.M.: Basics of Qualitative Research: Grounded Theory Procedures and Techniques. Sage Publications Inc., New Delhi (1990)
16. Suddaby, R.: From the editors: what grounded theory is not. Acad. Manag. J. **49**(4), 633–642 (2006)
17. Todd, P.M., Gigerenzer, G.: Ecological Rationality: Intelligence in the World. Oxford University Press, Oxford (2012)