A Cynefin based approach to process model tailoring and goal alignment

Marion Lepmets
Regulated Software Research Centre
Dundalk Institute of Technology
Dundalk, Ireland
marion.lepmets@dkit.ie

Aileen Cater-Steel
Faculty of Business, Education, Law and Arts
University of Southern Queensland
Toowoomba, Australia
aileen.cater-steel@usq.edu.au

Antoni Lluís Mesquida
Department of Mathematics and Computer Science
University of the Balearic Islands
Palma de Mallorca, Spain
antoni.mesquida@uib.es

Rory V. O'Connor
School of Computing
Dublin City University
Dublin, Ireland
roconnor@computing.dcu.ie

Tom McBride
Faculty of Engineering and IT
University of Technology, Sydney
Sydney, Australia
tom.mcbride@uts.edu.au

***Abstract*— In an industrial context all process models require a certain amount of tailoring to fit to the business environment of any specific organization in which the model is to be deployed. Process models should therefore be accompanied by tailoring guidelines and approaches to assist with strategic and operational goal alignment that support their use in industry. This paper explores shortcomings of process improvement and the existing process models, suggesting that a more holistic approach should be taken to process improvement in the modern organization. The paper provides an overview of systems thinking and the Cynefin framework that organizations can use to detect the characteristics of the domain in which they are operate. Knowing their domain helps the organization realize the amount of tailoring and goal alignment necessary to benefit from implementing process model guidance.**

Keywords—process improvement; process models; systems thinking; goal alignment; software development; IT service management; Cynefin.

I. INTRODUCTION

Successful process improvement supports the achievement of strategic and operational goals. In turn, process assessment helps organizations improve their ability to achieve their goals by identifying their critical process problems and establishing improvement priorities. There are a large number of process assessment and improvement models available, which we will refer to as process models in this paper. Among the most popular process models are CMMI, ISO/IEC 15504 and ITIL®. While there is an extensive amount of literature about the key success factors of process improvement, many improvements are not implemented or their benefits are not quantified. Additionally, there is little industry based research that evaluates the impact of these process models on achieving organizational goals or improving product qualities [1, 2]. In this paper we focus on formal process improvement, i.e. process improvements that follow model-based process assessments.

The motivation for this study is based upon a perceived lack of a holistic systems approach to software process improvement. All too often processes are assessed in isolation within an organization. The focus is on the improvement of a single process area without considering its impact on other processes, on the organization's business or product quality goals thus taking too narrow a view. This work is based on our previous research where we witnessed process improvement as isolated initiatives in both software development [3] and IT service management [1, 4, 5]. We propose taking a holistic approach where an organization is seen as a system, and where process improvement is but a part of a system within the larger organizational system. The process goals should be closely tied to the goals of the organizational system in order for the process improvement to become successful.

It has been observed that when it comes to the software development process, there is no universal process model suited to all situations [6]. The process models require a certain amount of tailoring in order to be applied beneficially to organizations. Most software organizations engage in the tailoring of standard software process models to their own particular operating context such as the size of the company, the target market, and project and system type [7]. The reason for this is that the process models themselves offer a generic solution and therefore require an approach to allow alignment between process goals and the organization's goals and situation. We claim that the amount of tailoring and alignment necessary depends on the organization's situation.

II. SYSTEMS THINKING AND THE CYNEFIN FRAMEWORK

Systems science argues that the only way to fully understand why a problem or element occurs and persists is to understand the parts in relation to the whole [8]. Systems thinking encourages understanding a system, i.e. any set or group of interdependent or temporally interacting parts, by examining the linkages and interactions between the elements that comprise the entirety of the system. In other words, systems thinking views problems as parts of an overall system, rather than reacting to specific parts, outcomes or events and potentially contributing to development of unintended consequences. Understanding the internal interactions requires integrating the components into something larger and more

capable than the components represent alone. In addition, a system resides within a comparable grouping of environmental factors that might also be called a context.

Kurtz and Snowden [9] proposed the Cynefin (pronounced Ku-nev-in) sense-making framework to help make sense of complex systems by explaining behaviors, decision-making and practices in terms of people's patterns of multiple experiences, personal, cultural and business based. The word "Cynefin" is a Welsh word that means "habitat", but more richly includes notions of the multiple experiences that people have in aspects of their lives. These experiences are a complex mixture of the personal, the wider cultural, and the business based or work place based. Cynefin is based on the notion that "humans use patterns to establish order in the world and make sense of things in complex situations". Cynefin originated in the practice of knowledge management with the aim of helping managers to "break out of old ways of thinking and to consider intractable problems in new ways" [9]. The Cynefin framework, illustrated in Figure 1, is a phenomenological framework, meaning that it is about how people perceive and make sense of situations in order to make decisions [9].

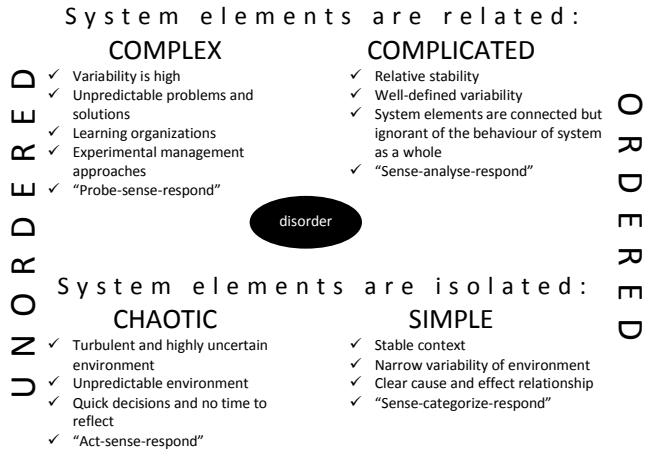

Figure 1 – Cynefin framework domains (adapted from [8])

The Cynefin framework has two large domains: order and unorder, each containing two smaller domains - simple and complicated in the ordered domain, and complex and chaotic in the unordered domain. In the centre of the framework is the fifth domain called disorder where multiple perspectives fight for prominence, factional leaders argue with one another and cacophony rules. Disorder should be avoided by organizations as it disrupts work. In the domain of order, the most important boundary of sense-making is between what we can use immediately (what is known) and what we need to spend time and energy on finding out (what is knowable). In the domain of unorder, distinctions of knowability are less important than distinctions of interaction; that is, distinctions between what we can pattern (what is complex) and what we need to stabilize in order for patterns to emerge (what is chaotic). In the ordered domain, the whole is the sum of the parts and the optimization of the system can be achieved by the optimization of the parts. In the domain of unorder, the whole is never the sum of the parts as any action changes the nature of the system. Cynefin's value as a sense-making framework lies in helping system decision-makers understand where their systems lie among these domains, and by extension, what kinds of tools, approaches, processes, or methods are more likely to work successfully in a given system [8].

III. THE ORGANIZATION'S DOMAIN

It is most important to understand that organizations live as whole systems, not as a collection of independent processes. And these systems exist in, and interact with an external environment that includes other systems as well as situational factors that can be irregular, highly variable and unpredictable [8]. A significant number of organizations' situations today qualify as complex. Their environment may change in short but irregular, unpredictable cycles, requiring the organization to adapt internally to avoid degradation. Decision-making processes depend on the situation.

In a simple situation, decision-makers sense, categorize and respond, i.e. they assess the facts of the situation, categorize them, and then base their response on the established practice [10]. Examples of this abound in standard procedures where all that is necessary is to decide what procedure to follow or to make other minor decisions within the procedure. The way incidents are handled by the Service Desk - received, then categorized and responded to - is an example of such a decision-making process. In the simple context there is no analysis of the impact customer satisfaction has on the entire software product or service system.

In the complicated context there are no established best practices that can be applied automatically, with little thought. Instead the decision-makers sense and analyze the facts to understand several options and their consequences on multiple levels, and finally respond. This can be observed in software development during the project planning phase when one or more domain experts consider the various stated or implied project goals and whatever is known about constraints, resources and risks before deciding how the project will be carried out. The decision-making process requires analysis, possibly by domain experts, indicating that the qualitative measures are gathered and analyzed before the decision-makers can respond to them. Also, the elements of the system should be related to each other. Ideally, understanding how process compliance and process efficiency relate to software product or IT service quality should be present here. This understanding could be achieved by mapping organizational goals, and product and service quality requirements to relevant software lifecycle or IT service management process goals. In this way, the decisions about what processes should be improved in order to achieve relevant organizational goals and/or increase certain software quality or IT service quality attributes can be made. From there, processes are tailored to achieve their planned outcomes using the available time and resources while addressing the known risks. In this way the complicated situation is addressed by considering the available information then adapting best practices to the situation.

Although most software and service organizations are complicated systems, failure to relate the system elements to each other indicates the assumption among the organizations that they operate as simple systems [1, 3].

In the complex domain, the decision-makers cannot impose a course of action but should allow the path forward to reveal itself while conducting experiments that are safe to fail. In other words, the decision-makers should probe first, then sense and finally respond. This can be observed when an organization forms a cross functional team to investigate and, if possible, derives an innovative solution to some situation that the organization's standard processes are inadequate to address. Possibly this is where dynamic capabilities [11] are achieving some success. Dynamic capabilities are "the organizational and strategic routines by which firms achieve new resource configurations as markets emerge, collide, split, evolve and die" [11]. Dynamic capabilities are simple (not complicated), experiential (not analytic), and iterative (not linear) processes. They rely on situation-specific knowledge that is applied in the context of simple boundary and priority setting rules. From Eisenhardt and Martin's description of dynamic capabilities it would seem reasonable to assume that dynamic capabilities emerge from the efforts of a cross functional team of domain experts who devised a process iteratively from several attempts to determine what would work in the specific situation. The result might be a process that works only for that situation or a process that can be adopted to use in similar future situations.

In the complex system, the decision-makers should constantly observe the environment to understand the dynamic forces around their organization. Here the decision-makers should also understand how the system elements affect the behavior of the entire system. Because there are no patterns here the best management approach is experimental, usually some form of "probe and learn". For example, in the event of a system or service failure the decision-makers need to probe into the system, observe its responses and analyze the cause of the failure from those responses. The possible ways forward will emerge from such an analysis. Also, evaluating the value co-creation with the customer helps the decision-makers to understand their environment. Mutual value creation stems from service logic and implies that all processes of a supplier that are relevant to its customer's business are coordinated with customer's corresponding processes into one integrated stream of actions. Understanding the entire ecosystem is a pre-requisite to managing business in a complex domain.

In the chaotic domain, the decision-makers must first act to establish order, then sense where the stability is present and where it is absent, and then respond by working to transform the situation from chaotic to complex where the emerging patterns can help prevent future crises.

Both simple and complicated contexts are heavily process-oriented, typically managed through the application of standard practice. In a complex context, problems and solutions emerge unpredictably that require a high degree of adaptive capacity. Organizations operating successfully in complex situations can also be called learning organizations [12] where the actors in the system are able to observe the impact of their initiatives and adjust accordingly to achieve the desired results.

Most software and service systems assume a complicated situation where the software developers and service providers attempt to standardize operations and make processes replicable. However, many software developers and service providers often look at their organizations in separate isolated units without seeing the entire system. This indicates an assumption of operating in the simple context where they collect the data about the isolated elements without understanding how these elements relate to one another.

IV. CYNEFIN AND THE SOFTWARE PROCESS PERSPECTIVE

The simple and complicated Cynefin domains require project leaders to adhere to a more fact-based management style. The simple domain is argued to be the domain of 'best practice' and is characterized by stability of the organisation and a clear cause-and-effect relationship, typically one in which the correct course of action or decision is often self-evident and undisputed, where all parties share an understanding that results in commonly agreed decisions.

The complicated Cynefin domains can be considered to be the domain of 'good practice' where there may be multiple competing appropriate solutions and where a clear relationship between cause and effect can be drawn. This domain requires expertise, investigating multiple options for possible software decisions. Here a project manager must not only listen to the advice of fellow team members but also embrace novel thoughts and solutions from others. This requires a willingness to experiment and often involves more creative approaches to enhance novel thinking and ultimately optimal solutions.

The complex domain is typically the area that causes the most difficulty for process improvement. Many software development issues fall into this category, where tacit knowledge ("Know How") is more important than explicit knowledge ("Know What") and adaptation of processes is necessary for success. In the delicate balance between process adherence and organisational structures, it is in this complex domain that recognition of starting point and appreciation for emergent order is key for positive outcome.

Both simple and complicated domains are heavily process oriented where the guidance of the process models has potential for the most benefit. While in the simple domain, the process model guidance may be considered sufficient to tackle a situation, the complicated domain requires additional goal alignment to maximize benefit from the process model guidance. There are several studies conducted in this area of aligning process goals to organization's business and product goals [1, 3, 13-17] but there is still no comprehensive approach that industry has embraced for goal alignment in process improvement.

The complex domain presents the biggest challenge for process models. This domain is characterized by synergy of people, open-mindedness and innovativeness in problem-solving, and goal internalization in decision-making which process models do not cover. While agile development

methodologies and Scrum project management might be best suited to the complex domain in software development so far, there is little in the way of explicit guidance for iterative process improvement [18]. This issue has also not been widely recognized in industry. We suggest that existing process models, as they stand today, are not suited to the complex Cynefin domain. However, the majority of organizations today are operating in complex situations. Therefore a significant issue to be addressed is what type of process models can provide a solution and if any amount of tailoring of the existing process models might be enough? So far there have been only few attempts to study dynamic capabilities in software development so this could be a fruitful area for research.

V. Conclusions and Future Works

We suggest organizations align their process goals with organization goals in order to benefit from the improvements made as well as to realize these benefits. We also recommend multiple methods to measure process performance and efficiency to realize the impact of process improvement on organizations' business, project and product goals.

We recommend the use of the Cynefin framework to understand the domain the organization inhabits. The knowledge about the domain dictates the amount of tailoring and goal alignment necessary for successful implementation of process model guidance in the organization. More work should, however, be conducted to understand if and how the process models can be applied in the complex domain.

References

[1] M. Lepmets, A. L. Mesquida, A. Cater-Steel, A. Mas, and E. Ras, "The Evaluation of the IT Service Quality Measurement Framework in Industry," *Global Journal of Flexible Systems Management,* vol. 15, pp. 39-57, 2014.

[2] B. Barafort, V. Betry, S. Cortina, M. Picard, M. St-Jean, A. Renault*, et al.*, *ITSM Process Assessment Supporting ITIL*. Amersfoort: Van Haren Publishing, 2009.

[3] M. Lepmets, T. McBride, and E. Ras, "Goal Alignment in Process Improvement," *Journal of Systems and Software,* vol. 85, pp. 1440-1452, 2012.

[4] M. Lepmets, E. Ras, and A. Renault, "Impact Analysis of Process Improvement on IT Service Quality," presented at the International Conference on Exploring Service Science (IESS), Geneva, Switzerland, 2011.

[5] A. L. Mesquida, A. Mas, E. Amengual, and J. A. Calvo-Manzano, "IT Service Management Process Improvement based on ISO/IEC 15504: A systematic review," *Information and Software Technology,* vol. 54, pp. 239-247, 2012.

[6] B. Boehm and R. Turner, *Balancing Agility and Discipline - A Guide for the Perplexed*. Boston 2004.

[7] G. Coleman and R. O'Connor, "Investigating software process in practice: A grounded theory perspective.," *Journal of Systems and Software,* vol. 81, pp. 772-784, 2008.

[8] H. W. Dettmer, "Systems Thinking and the Cynefin Framework - A Strategic Approach to Managing Complex Systems," 2011.

[9] C. F. Kurtz and D. J. Snowden, "The new dynamics of strategy: Sense-making in a complex and complicated world," *IBM Systems Journal,* vol. 42, pp. 462-483, 2003.

[10] D. J. Snowden and M. E. Boone, "A Leader's Framework for Decision Making," *Harvard Business Review,* vol. November 2007, pp. 69-76, 2007.

[11] K. M. Eisenhardt and J. A. Martin, "Dynamic Capabilities: What Are They," *Strategic Management Journal,* vol. 21, pp. 1105-1121, 2000.

[12] P. M. Senge, *The Fifth Discipline: the Art and Practice of the Learning Organization*. London: Random House, 1990.

[13] P. Clarke and R. V. O'Connor, "The situational factors that affect the software development process: Towards a comprehensive reference framework," *Information and Software Technology,* vol. 54, pp. 433-447, 2012.

[14] T. McBride and M. Lepmets, "The Many Forms of Process Improvement - Results of an International Survey," presented at the Software Quality Days, Vienna, Austria, 2012.

[15] A. Shrestha, A. Cater-Steel, M. Toleman, and W.-G. Tan, "A Decision Support Tool to Define Scope in IT Service Management Process Assessment and Improvement," presented at the DESRIST 2013, Helsinki, Finland, 2013.

[16] S. Jeners, R. O'Connor, P. Clarke, H. Lichter, M. Lepmets, and L. Buglione, "Harnessing Software Development Contexts to Inform Software Process Selection Decisions," *Software Quality Professional* vol. 16, pp. 35-46, 2013.

[17] F. Stallinger, R. Neumann, R. Schossleitner, and R. Zeilinger, "Linking Software Life Cycle Activities with Product Strategy and Economics: Extending ISO/IEC 12207 with Product Management Best Practices," in *SPICE'2011*, Dublin, 2011, pp. 157-168.

[18] O. Salo and P. Abrahamsson, "An Iterative Improvement Process for Agile Software Development," *Software Process: Improvement and Practice,* vol. 12, pp. 81-100, 2007.