Can Working Style Be Identified?

Marcello Sarini[1]

University of Milano-Bicocca, Department of Psychology,
I-20126 Milan, Italy. marcello.sarini@unimib.it

Abstract. Working style is a concept not fully taken into account especially when the unfolding of a business process is supported by Process-aware Information Systems. In this paper we present a practical approach to define and identify working style. The identification is done by searching for some patterns representing critical common features which define a working style. A prototype is presented as a proof of concepts for this new research path.

Key words: Working style, Organizational mining, Graph databases, Process-aware Information Systems, Visualization.

1 Introduction

The concept of style has been widely discussed in different fields such as Arts, Fashion, and Design. Research on style focused on two different complementary perspectives, on the one hand on the *artifacts* which are characterized by a certain style; on the other hand on the *process* of designing the artifacts [8]. In this last perspective, more related to the Design field, style is an organizing principle emerging from a series of choices made within some set of constraints [6, 21]. In our proposal we want to fill this gap by creating an artifact, the *Working Style Artifact* (WSA), which reifies the way people work by making visible the interdependencies among performers who completed their work related to a business process. Even if little has been told about the concept of style associated to the way people work, *Working style* considers the process of work as a set of choices made by the involved performers within the set of constraints posed by the workplace; and this is crucial especially when work is performed with the support of Process-aware Information Systems as these systems tend to pose strong constraints on the way work is arranged during the unfolding of the supported business process.

In this view, the concept of Working style, and especially its visible counterpart, the Working Style Artifact, can be seen as a tool for facilitate comparisons to identify similarities and differences about the nature of work; this could be useful in different situations where it would be meaningful to make visible and commonly acknowledged in a seamless way to the involved stakeholders repetitions of work behaviors. This seems relevant for instance in different areas of Job analysis where Job analysis is the process by which jobs are subdivided into elements, such as tasks, through the application of a formalized, systematic

procedure for data collection, analysis and synthesis [14]: e.g., to promote *organizational compliance*, i.e., to facilitate the comparison of real work performed by an actor with the tasks expected to be completed according to her specific position; to facilitate *team formation*, by choosing people who are similar in working [20]; to limit the possible raise of *resistance to change*, by facilitating the comparison of work performed during the as-is process with the tasks expected for the redesigned to-be process. All of these aspects will be crucial for a well functioning organization, as organizational compliance is about the capability of an organization of setting clear duties for a determined position; this is the opposite of role ambiguity, where studies in organizational Psychology emphasized that role ambiguity could negatively affect, among the others, job satisfaction [19]. It would be useful to keep into account that working style could be associated to the concept of habits in work settings and changing habits could be troublesome [27], leading also to resistance to change phenomenon which often hampers redesign efforts [12]. All in all, taking into account working style would also lead to the satisfaction of both joint understanding and involvement principles advocated as two of the identified ten principles for good business process management [25].

Research on style is interested to answer different critical questions about style, and how a style could be identified is one of them [21]. The visual identification of a style would follow two different theoretical approaches as described in [15]: *template matching* and *feature analysis*: the first approach relies on the possibility to compare each new input with a standard; the last ones focuses on the identification of a common set of critical features which characterizes the style itself [7]. Here we propose a similar approach to the definition and identification of a working style focusing on a set of features describing possible aspects of working style as reported within the Working Style Artifact. For the first step towards its identification, we consider working style in terms of *handover-orientedness* and *rework-orientedness*. The first feature characterizes the working style of a person in terms of level of handovers and this would relate to the degree of interdependence in the work of a performer. The second feature is intended to characterize the working style of a person in terms of amount of rework and this is strictly connected to the concept of efficiency in working or, following the idea of waste reduction in Lean thinking [16], to the reduction of over-processing. To give these features a concrete meaning we associate to them a set of *patterns* identifying the repetition of a specific work condition charactering the style of work in terms of the considered features which should be then made visible and recognizable within the corresponding Working Style Artifact.

In Computer Science the concept of pattern was mainly exploited in the Software Engineering field, focusing on design patterns [9], intended as tools helpful to facilitate the reuse of object oriented software already coded for a similar problem. The idea of the authors is rooted in the work of Alexander [4], an Architect, who defined patterns as "elements which are a collective memory of things which work in our surroundings". Lately, the concept of design pattern was extended also to the Business Process Management area, with the definition of

workflow patterns [24], built with the same rationale in mind of design patterns for Software Engineering. We considered here patterns, and more specifically work patterns, concerning a repetition of obversable behaviors [1]. From the perspective of identifying patterns as a repetion of a structure, and in particular referring to the definition of working style, we took strong inspiration from the field of Organizational mining. Organizational mining was defined by [20] as part of the broader Process Mining area where the focus is on the possibility to extract any meaningful repetition of behaviors regarding the organization of work from logs of real work. Similarly, patterns of work has been identified in terms of Affordance Networks [18] within Organizational Routines, where again the concept of organizational routine is related to "repetitive, recognizable patterns of action" [17].

In the following of the paper we describe our approach to identify working style by means of Working Style Artifacts; Conclusions and Future Works conclude the paper.

2 The Proposed Approach

As the presence of Process-aware Information Systems has become readily available, it is assumed that it is possible to extract information from transaction logs such as i) events concerning an activity (a well defined step in a process); ii) the case to which an event is related (the istance of the process); iii) the performer of the event; iv) and the total order of events [23]. In this way, the nature of work would be analyzed thanks to this information. From this premise, we propose a prototype which allows users to search for patterns where the identified patterns would be related to features describing some characteristics of a working style. Patterns will be searched within the Working Style Artifact by means of a *pattern search interface* (See e.g., Fig. 4). For the search of patterns we took inspiration from [23] by considering the metrics proposed, in particular the metrics based on the identification of possible causality of work such as handovers. The occurences of the eventually found patterns are then visualized directly in the Working Style Artifact. In addition, to facilitate the identification of the occurrences of the selected patterns, we considered to support *juxtaposition* [10] which is a technique for visually comparing information which aims at helping "the users shift their attention between objects or see patterns between elements" (compare e.g., the initial Working Style Artifact in Fig. 3 with the Artifact displayed in Fig. 5 after the highlighting of the found patterns).

The prototype is built using Python programming language [3] togheter with Neo4j, a graph database [2]. The choice of this database was driven by the fact that it provides a query language, Cypher, which is oriented to the identification of patterns within graphs. Moreover, the Cypher language, was already used for its flexibility in the field of Business Process Management [13]. For what concerns the idea of visualizing patterns of work behavior, we took inspiration from another tool, the EventExplorer by [5], used to identify and visualize patterns of events regarding consumers' activities during online commerce sessions.

2.1 The Identification of Patterns

To illustrate our prototype we took inspiration from the log presented in [20]. The prototype first reads the log data expressed in CSV format (see Fig. 1), where in each row is reported a case, an activity performed, a performer and a timestamp. At this point, the CSV data are transformed into the related db graph. This would then facilitate the search of patterns within Working Style Artifact through the functionalities provided by the Cypher language.

1	A	John	1
1	B	Mike	2
1	C	John	3
1	D	Sue	4
1	E	Pete	5
1	F	Jane	6
1	H	Sue	7
1	A	John	8

Fig. 1. An excerpt of the csv log from case 1.

The graph db will include nodes: `Cases`, `Activities`, and `Performers` and relations: a Case `includes` an Activity, an Activity is `performed_by` a performer, and most important of all, for the search of patterns related to working style, a Performer `works_with` a Performer, where this relation is set into the database whenever a Performer p_1 performed an activity immediately before an activity performed by p_2. As it is possible to notice in Fig. 1, John performs activity A before Mike performs B, and this is represented in the graph database with the creation of a relation `works_with` connecting John to Mike (see Fig. 2).

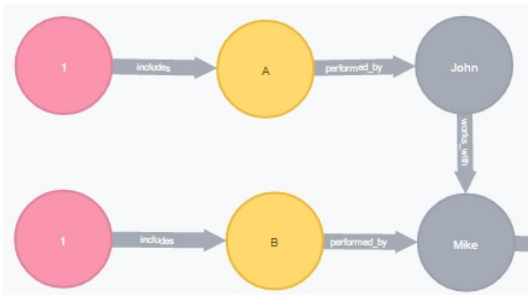

Fig. 2. An excerpt of the db graph generated from case 1.

From the graph generated, we are able then to build the corresponding Working Style Artifact (see Fig. 3) which is structured in this way: each row represents a case, while each cell in the rows is colored as to represent the related performer in completing an activity for that case. For instance, John in the first case performs both the first and the third activity (see Fig. 1) and consequently both the

first and the third cell of the first row of the Working Style Artifact are colored in red which is assumed to be the color representing the performer John within the Working Style Artifact (see Fig. 3).

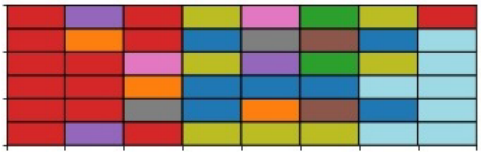

Fig. 3. The Working Style Artifact.

The search for the patterns of interest is specified through a Cypher query. In this query are to be specified: i) the two performers at the edge of the pattern (both the request for a specific performer or for any performer is allowed); ii) the length of the path connecting the two performers (with the Cypher language different options are possible, such as to search for paths of any length, to search for paths with a specific length, with at least or at most a specific length, or to search for paths with a length falling into an interval); iii) if any performer is selected at one edge of the pattern, whether to search for patterns including only the same or different performers; iv) whether the activities performed by the two performers have to be the same or not. To facilitate the definition of the query to search for a pattern we designed a pattern search interface generating automatically the desired query.

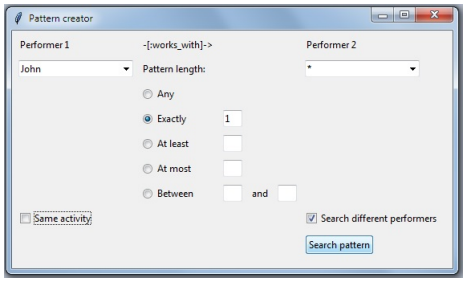

Fig. 4. The pattern search interface for handover-orientedness.

In Fig. 4 for instance it is displayed how to set the user interface to search for handover patterns in which John is the first performer completing an activity (p_1 is selected as "John" and p_2 as "any"), and there is a direct handover of work with other performers (path length is set to "Exactly 1" and also the "search different performers" check box is set). The results of the query both identifies some metrics counting the occurrence of the pattern, and identifies the recurring patterns which are then visualized accordingly in the Working Style Artifact by

coloring in light blue all the cells corresponding to the performers involved in the patterns (See Fig. 5).

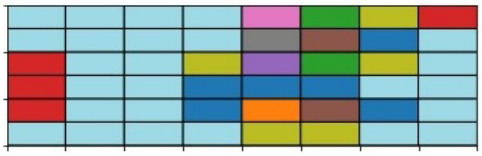

Fig. 5. The WSA after the identification of patterns for handover-orientedness.

Another interesting example is related to the identification of some forms of rework for a performer. The idea is to look for patterns whenever a performer, e.g. John, is involved into some rework, (see Fig. 6).

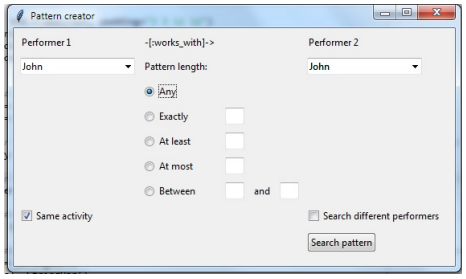

Fig. 6. The pattern search interface for rework-orientedness.

In this case the specification of rework is done by selecting "John" for both performers and selecting "Same activity" with paths of any length. The occurrences of the pattern are then visualized in the Working Style Artifact (See Fig. 7).

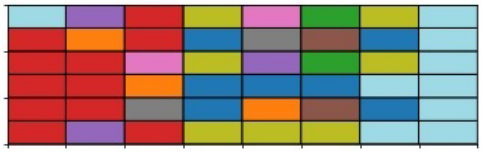

Fig. 7. The WSA after the identification of patterns for rework-orientedness.

The first pattern considered in the examples is about the feature of handover-orientedness to characterize working style. In the presented case, John's working style is handover-oriented as it is characterized by the presence of the handover pattern in all of the considered cases (as it is possible to notice by comparing the Working Style Artifact in Fig. 3 with the Working Style Artifact after the

search of the handover patterns as displayed in Fig. 5). In the second example (see Fig. 7), the pattern we looked for is related to rework-orientedness feature. In the presented case, the rework pattern is identified with just one occurrence only in the first case, so due to its scarse presence, John's working style is not to be considered rework oriented.

3 Conclusions and Future Works

In this paper it is described our approach to identify working style through the definition of a Working Style Artifact. This Artifact reifies visually the way people work according to the data contained in a log file related to the activities they performed in a business process. It is explained how to build and query the Working Style Artifact by means of a prototype implemented using the Python programming language [3] and the Neo4j graph database [2]. Working style is determined by searching within the Working Style Artifact for the occurrence of some patterns which characterize some critical common features of the working style. However, other features could be considered to enrich the set of common features characterizing a working style e.g., feature of autonomy: whether a performer is the only one to perform the activities of a case or not; feature of redundancy: whether a performer in any single case is replaceable or not by other performers in the completion of the same activities; feature of question-answers among pair of performers reflecting the occurrence of some Activity Patterns [22]. Future works will be oriented to apply the approach to real cases by considering a richer set of common features to characterize a style. Then, the future prototype will aim at facilitating both the visual pattern recognition associated to the features of the style and the visual comparison of Working Style Artifacts by considering additional visualization techniques, such as Arc Diagrams [26] and Rythm-Eye views [11]. These methods could be more useful than the simpler visualization of the Working Style Artifact proposed in the prototype. In fact, the first method is aimed at making more explicit the relations among different segments of the considered Working Style Artifact so to facilitate the identification of patterns, while the second method could make the sequences in the Working Style Artifacts more easy to be compared as the presentation of them is not linear and dispersed but built around a circle to facilitate comparison at a glance. This would be especially useful for visually comparing larger Working Style Artifacts.

References

1. Merriam-webster. *https://www.merriam-webster.com/dictionary/pattern.*
2. The Neo4j graph database. *https://neo4j.com/.*
3. Python programming language. *https://www.python.org/.*
4. C. Alexander. *Notes on the Synthesis of Form.* Harvard University Press, 1964.

5. P. Bodesinsky, B. Alsallakh, T. Gschwandtner, and S. Miksch. Exploration and Assessment of Event Data. In *EuroVA Workshop*, 2015.
6. C.-S. Chan. Exploring Individual Style in Design. *Environment and Planning B: Planning and Design*, 19(5):503–523, 1992.
7. C.-S. Chan. Can Style Be Measured? *Design Studies*, 21:277–291, 2000.
8. C. M. Eckert and E. Y. Do. Special Issue: Understanding, Representing, and Reasoning about Style. *AI EDAM*, 20(3):163–165, 2006.
9. E. Gamma, R. Helm, R. Johnson, and J. Vlissides. *Design Patterns: Elements of Reusable Object-oriented Software.* Addison-Wesley, 1995.
10. M. Gleicher, D. Albers, R. Walker, I. Jusufi, C. D. Hansen, and J. C. Roberts. Visual Comparison for Information Visualization. *Information Visualization*, 10(4):289–309, Oct. 2011.
11. J. Gulden. *Visually Comparing Process Dynamics with Rhythm-Eye Views*, pages 474–485. Springer International Publishing, Cham, 2017.
12. R. Hirschheim and M. Newman. Information Systems and User Resistance: Theory and Practice. *Comput. J.*, 31(5):398–408, Oct. 1988.
13. K. Kammerer, J. Kolb, and M. Reichert. *PQL - A Descriptive Language for Querying, Abstracting and Changing Process Models*, pages 135–150. Springer International Publishing, 2015.
14. E. McCormick. *Job and Task Analysis*, pages 651–96. Chicago: Rand Mc nalley, 1976.
15. U. Neisser. *Cognitive Psychology.* Appleton-Crofts, New York, 1967.
16. T. Ohno. *Toyota Production System: Beyond Large-Scale Production.* Productivity, 1988.
17. B. T. Pentland and T. Hærem. Organizational Routines as Patterns of Action: Implications for Organizational Behavior. *Annu. Rev. Organ. Psychol. Organ. Behav.*, 2(1):465–487, 2015.
18. B. T. Pentland, J. Recker, and G. M. Wyner. A Thermometer for Interdependence: Exploring Patterns of Interdependence Using Networks of Affordances. In *Proceedings of ICIS 2015*, Dec. 2015,
19. J. R. Rizzo, R. J. House, and S. I. Lirtzman. Role Conflict and Ambiguity in Complex Organizations. *Administrative Science Quarterly*, 15(2):150 – 163, 1970.
20. M. Song and W. M. P. van der Aalst. Towards Comprehensive Support for Organizational Mining. *Decis. Support Syst.*, 46(1):300–317, Dec. 2008.
21. M. Stacey. Psychological Challenges for the Analysis of Style. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 20(3):16784, 2006.
22. L. Thom, M. Reichert, and C. Iochpe. Activity Patterns in Process-aware Information Systems: Basic Concepts and Empirical Evidence. *Int. J. Bus. Process. Integrat. Manag. (IJBPIM)*, 4(2):93–110, 2009.
23. W. M. P. van der Aalst, H. A. Reijers, and M. Song. Discovering Social Networks from Event Logs. *Computer Supported Cooperative Work (CSCW)*, 14(6):549–593, 2005.
24. W. M. P. Van Der Aalst, A. H. M. Ter Hofstede, B. Kiepuszewski, and A. P. Barros. Workflow Patterns. *Distrib. Parallel Databases*, 14(1):5–51, July 2003.
25. J. vom Brocke, T. Schmiedel, J. Recker, P. Trkman, W. Mertens, and S. Viaene. Ten Principles of Good Business Process Management. *Business Process Management Journal*, 20(4):530–548, 2014.
26. M. Wattenberg. Arc Diagrams: Visualizing Structure in Strings. In *IEEE Symposium on Information Visualization, 2002. INFOVIS 2002.*, pages 110–116, 2002.
27. W. Wood, J. Quinn, and D. Kashy. Habits in Everyday Life: Thought, Emotion, and Action. *Journal of Personality and Social Psychology*, 83 (6):1281–1297, 2002.