Business Process Completeness

Foundation of Business Knowledge Management

Shuichiro Yamamoto
Information Engineering, IPUT in Nagoya
Nagoya, Japan
e-mail: yamamoto.shu@n.iput.ac.jp

***Abstract*—A clear definition of business processes is required to realize business. A business process is only complete when the problem is addressed. Additionally, it is difficult to address a problem if it is not identified. In this paper, we propose a comprehensive business process completeness concept from the aspects of business process, consisting of business process acceptance/resource/judgement conditions and exceptions that propagate between processes. In addition, a self-process completeness diagram is proposed to analyze the comprehensive process completeness. Furthermore, we confirm the effectiveness of the proposed method using examples.**

Keywords-business process management; knowledge transfer; Self-Process Completeness Diagram.

I. INTRODUCTION

In Japan, several inspection test frauds of manufacturing industry have recently been discovered and have become social problems [1]. The top management of a Japanese automobile company apologized for the inspection test fraud, saying, "We may have misjudged the workload." It is clear that, if a company accepts orders that exceed its production capacity, it will not be able to produce the required amounts of products or services, or even if it is able to produce them, the quality of the products or services will degrade.

If an organization does not know its production capacity, it cannot know when the number of orders exceeds its production capacity. The production capacity at the time of planning often falls below the organization's planned production capacity at the time of execution due to excessive orders or changes in materials required for production. Not everything goes according to plan. Therefore, it is necessary to design business processes that can detect deviations from the plan as exceptions and respond to them. Conversely, if the upper limit of production capacity is known, it is possible to limit further orders by detecting an excessive number of orders as an exception. In order to correctly execute a business process, it is necessary to know the execution capability of the business process. Therefore, it is important to correctly define and confirm not only business process but also process execution conditions.

In this paper, we propose the Self-Process Complete Diagram (SPCD) as a model for designing the production process in industry and clarify that it can be applied to manage process completeness. Below, Section II describes related research. Next, Section III proposes SPCD as a means to manage comprehensive completeness among whole production processes. Section IV describes an application example of SPCD. In Section V, we discuss our considerations, and in Section VI, we present the conclusion.

II. RELATED WORK

Related studies on Ji-Koutei-Kanketsu (JKK), Knowledge transfer, Business Process Modeling (BPM), Self-Organized Process, and Functional Resonance Analysis Method (FRAM) are explained below.

A. Ji Koutei Kanketsu

In the production process, there is a misconception that local optimization is necessary, as long as one's own process is fine and that unnecessary problems shall not be introduced to one's own department. If a problem is discovered at the final stage of development, the design cannot be modified or the basic structure of the product cannot be changed. Therefore, comprehensive product design and manufacturing is required throughout the entire production process. Ji-Koutei-Kanketsu (JKK) is a method that optimizes the entire production process, not just a specific process. The Japanese words Ji, Koutei, and Kanketsu [2] are self, process, and completion, respectively.

To introduce JKK, it is necessary to define not only business procedures that define the flow of work, but also requirements organization sheets that define business requirements. The requirements organization sheet consists of fields of the necessary items/information, business inputs, and business outputs for each business process. The necessary item and information field clarifies the input, tools, methods, capabilities/authority, and reasons as conditions for the quality of product. The input field describes the receiving criteria, such as when, where, and what. The output field describes where to sink, by when, and what to produce. The criteria field describes criteria for determining that "the output of the process is good."

JKK's production processes can also be seen as business processes. JKK clarifies the completeness conditions for each business process element. The requirement organization sheet is an essential feature of JKK.

B. Knowledge transfer

In order to transfer a company's experiential knowledge, it is necessary to clarify business processes. For this reason,

 ISBN: 978-1-68558-165-7

methods for clarifying business processes have been proposed for knowledge transfer.

From a knowledge perspective, processes need to be defined to provide appropriate knowledge for tasks in an organization's operational business processes. In addition, knowledge must be extracted for the long-term growth, development, and competitiveness of companies. However, unless valuable knowledge within an organization is externalized or formalized, it cannot be used by other employees and disappears from the company. Therefore, Knowledge management shall be established using Business Process Modeling (BPM). Salvadorinhoa and Teixeira [3] pointed that BPM can not only help organizations improve their Industry 4.0 environment, but also facilitate knowledge acquisition and distribution.

C. *Business Process Modeling*

Ore et al. [4] proposed a Self-managed organization based on Business Process Management. They showed a need for the business process management approach, which would manage the need for keeping critical business processes continuity and self-managed way of working of autonomous teams.

As long as the digitalization of business is promoted, business process documentation becomes vital for business process continuity. The digitalization re-constructs the traditional business processes into a new digitalized business processes [5]. For example, Digital Balanced Scorecard (DBSC) [6] consists of digital business processes.

There are many Business Model notations including Business Process Models. Yamamoto [7] compared the representation capability of Business Model notations by defining fifteen key features of these notations with five interrogatives.

Leonard and Swap [8] defined deep smart as the expertise that allows experts to instantly grasp complex situations and make quick and wise decisions in order to deal with real problems. That is, deep smart is “strong expertise formed by beliefs and social influences that can generate insights based on tacit knowledge grounded in direct experience.” For example, in production process design, the problem is how to transfer defect investigation knowledge from experienced workers to beginners. An example of deep smart is the failure investigation knowledge that experienced engineers have. Leonard and Swap pointed out the importance of acquiring empirical knowledge through experimental learning. However, no concrete experimental learning method has been clarified. In addition, they have not clarified the knowledge representation of deep smarts. If deep smart cannot be expressed, it remains tacit knowledge, and deep smart knowledge transfer from experts to beginners is individual and difficult to spread horizontally.

As a technique for improving production processes in the manufacturing industry, Mono-Koto-Bunseki (MKB) (in Japanese) has been proposed [9]. Mono, Koto, and Bunseki mean Entity, Process, and Analysis, respectively. By treating objects such as materials and products as “entities” and the series of activities that make products from materials as "process," MKB can analyze the production process, discover waste, and optimize it.

Yamamoto and Fujimoto [10] proposed the Production Knowledge Chart (PKC) that expresses the production process to acquire the empirical knowledge necessary for investigating defects in manufacturing processes.

Object Process Methodology (OPM) proposed by Dori includes Object and Process [11][12]. For example, the aircraft design OPM has a Stakeholder Needs Set, Assumptions and Constraints Sets, and Requirements as Objects. There are three types of Processes: Defining, Realizing, and Implementing. In addition, physical Objects include Aircraft, System, Item, and Item component.

D. *Self-Organized process*

Bussmann and Schild [13] developed a strictly decentralized approach to manufacturing control by using workpiece and machine agents. Machine agents manage a virtual buffer. Workpiece agents manage the state of workpieces. They showed a capacity bottleneck is automatically propagated in the opposite direction of the material flow.

Graessler et al. [14] clarified the process changes and opportunities for the development process by the vision of Self-Organizing Production Systems (SOPS). Main features of SOPS are as follows. SOPS consists of segmented autonomous modules instead of one connected system. Distributed control procedures of SOPS manage to react to unexpected changes of the production system. Connecting to related services and devices allows them to exchange information regarding the execution of their own production processes.

E. *Functional Resonance Analysis Method*

Functional Resonance Analysis Method (FRAM) [15] has been used to analyze complex functional resonances of socio-technical systems through functional networks. The FRAM function is defined by hexagonal nodes with six sides. These sides correspond to six aspects which are Input, Output, Time, Control, Resource, and Precondition. The output side of a function can be connected to the other five sides of other functions. FRAM provides useful means for safety analysis. Possible aspect relationships are <O, I>, <O, T>, <O, C>, <O, R>, and <O, P>. Here, <X, Y> is where X and Y are functional aspects.

The following three types of FRAM matrix representations have been proposed.

Lundberg and Woltjer [16] proposed a Resilience Analysis Matrix (RAM) to visualize functional dependencies between complex systems. RAM is a square matrix that shows the propagation relationship between functions. The size of RAM is the number of functions in FRAM. Element (i, j) of RAM indicates that some aspect of function i is propagated from the output of function j. The diagonal element (i, i) of RAM is the output of function x.

Patriarca et al. [17] proposed another square matrix composed of aspect combinations of FRAM functions. If there are n couplings in FRAM, RAM is defined as an n × n square matrix. The value of RAM (i, j) is 1 or 0.

Functional Aspect Resonance Matrix (FARM) is a non-square matrix that shows the propagation relationship between the output of a function and other aspects [18]. The number of rows in FARM is the number of output sides of the function that are propagated to other functions in FRAM. The column size of FARM is the number of sides of a function that are connected from the output sides of other functions. Element (i, j) of FARM indicates that some functional surface j is propagated from the output of function i. In general, the number of rows and columns in FARM are not equal, so there are no diagonal elements. The equivalence of the above three matrices has been shown by Yamamoto [18].

III. SELF COMPLETE BUSINESS PROCESS

A. *Self-Process Complete Diagram*

Self-Process Complete Diagram (SPCD) is defined by hexagonal nodes with six sides. These sides correspond to six aspects which are Input, Output, Acceptance condition, Resource condition, Exception condition, and Judgement condition. The acceptance, input, resource, and judgement aspects represent outside-in flows from external elements. The output and exception aspects represent inside-out flows to external elements.

Figure 1 shows an example of SPCD.

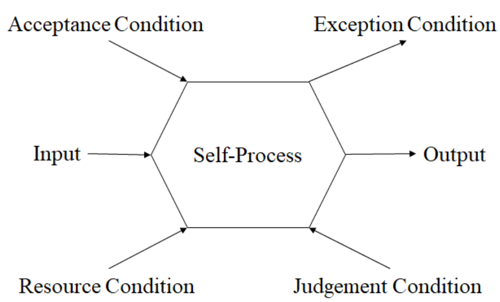

Figure 1. Example of Self-Process Complete Diagram.

The metamodel of SPCD is shown in Figure 2. There are two relationships, i.e., connection and propagation relationship.

The connection relationship defines the binary relationship that flows from the output aspect of a process into the input aspect of other processes. The connection relationship is used to define business process flows.

The propagation relationship defines 1) the exception condition of a process flows into acceptance condition of other process, and 2) the exception condition of a process flows into the exception condition of other processes.

The propagation relationship is used to propagate exceptions of a process into forwardly and backwardly other processes.

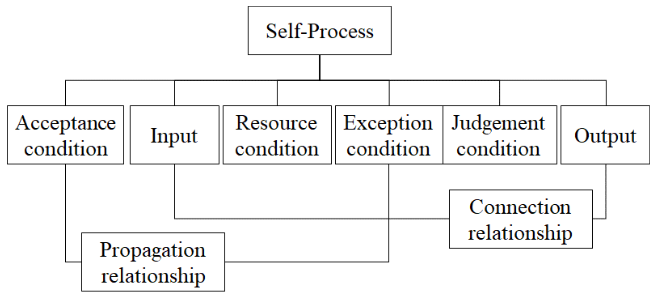

Figure 2. Metamodel of Self-process Complete Diagram

Figure 3 shows an example of propagation relationship. In Figure 3, there are two processes of a production plan and a production for delivery. The production plan process accepts a purpose of plan and generate the production order. If the production for delivery process accepts the production order, then it generates the product. In case of production capacity is not sufficient, the production delay occurs as the exception in the process. The exception is propagated to the acceptance aspect of former process. Then the former process is noticed that the purpose of the plan is no more realized.

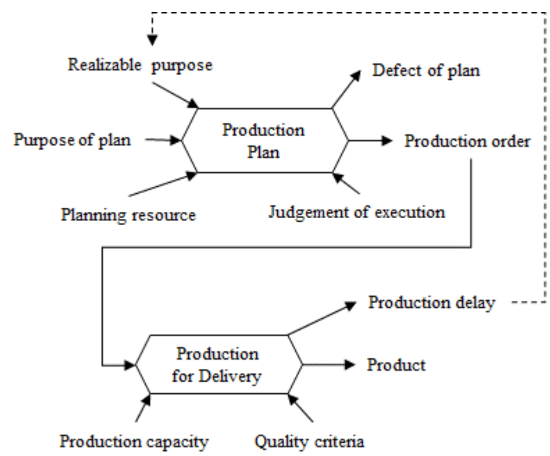

Figure 3. Example of Self-processes for production

B. *Conditions of the Complete Self-Process*

The following are conditions used to check if the process itself is complete.

If the acceptance conditions are not met, the process will not start.

Unless the resource conditions are met, the process will not start.

If the result of the own process does not satisfy the judgment conditions, it will not be output.

Generates an exception condition when the own process cannot start or when the output does not satisfy the judgment conditions.

When the resource conditions are satisfied for the input that satisfies the acceptance condition, generate an output that satisfies the judgment condition of the own process.

C. *Business Process Analysis with SPCD*

Business process analysis using SPCD is as follows.

[step1] Describe business processes and flows among processes with input and output arrows.

[Step2] Clarify resource, acceptance, and judgement conditions for each process.

[Step3] Analyze the possibility of deviations under above three conditions of each process.

[Step4] Identify exceptions of each process based on deviations analyzed.

[Step5] Analyze propagations of exceptions among processes.

There are two directions of propagation: upward and downward propagation. The upward propagation feeds back exceptions from a downstream business process to its upstream business processes. The downward propagation feeds exceptions from an upstream business process to its downstream business processes.

The exception propagation analysis is used to discover candidates of business process improvement. If an exception is notified to a process, the process should change acceptance, resource, and judgement conditions to handle the exception. This presents an opportunity for process evolution to improve sustainability in response to environmental changes.

IV. CASE STUDY

In this section, two case studies are explained to show the applicability of SPCD.

A. *Alcoholic beverage delivery*

When I went on a trip to a north region of Japan, I decided to buy two bottles of sake from that region at a local liquor store and send them home. By paying for the local sake and filled out the delivery slip, I asked a courier to send the sake packaged by the liquor store to the address on the slip. A courier delivered the package to a distribution center near my home. At the distribution center, they noticed that local sake was leaking. There was a problem with the sake during delivery, so the distribution center requested the sending liquor store to repack it and redeliver it. Due to sufficient packaging, the two bottles of sake were re-delivered safely. Figure 4 shows the flow of these processes along with backpropagation of exceptions.

The dotted lines show propagations of exceptions. For example, the “insufficient packaging” exception in “Packed for local delivery” process propagates to the acceptance condition of “Receive alcoholic beverages” process. Then the “Alcohol is leaking” exception occurred in “Receive alcoholic beverages” process. The exception again propagates to the acceptance condition of “Packed for local delivery” process.

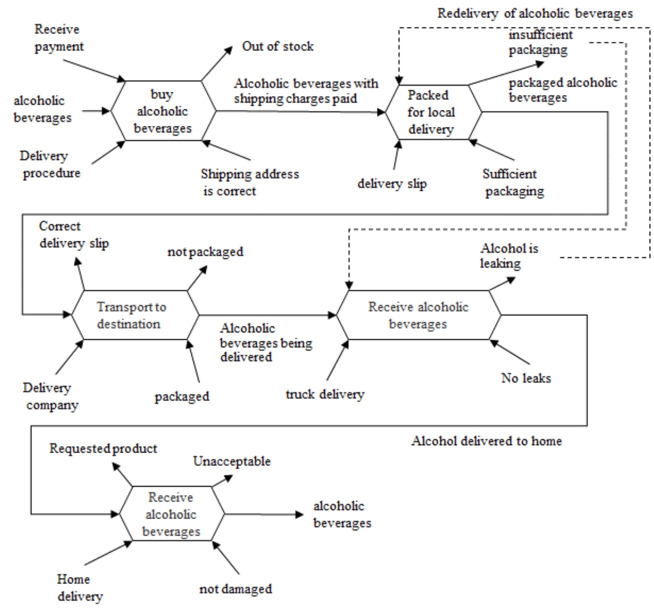

Figure 4. Example of Alcohol beverage delivery.

B. *Strawberry cake shipping*

There was an online sale in which strawberry cakes for Christmas were delivered on Christmas Eve. This strawberry cake was supervised by a famous pastry chef and became popular, with many orders placed. However, due to the intense summer heat, the strawberry crop failed, and they were unable to procure the strawberries they needed right away. Production of the cake was delayed due to a delay in the procurement of strawberries. Furthermore, during the shipping process, the manufactured cake had to be frozen for a certain period of time to maintain quality. As a result, delays in the procurement of strawberries caused delays in production and insufficient freezing time. As a result, some cakes collapsed when delivered to consumers.

Figure 5 shows the result of describing this process flow from order reception to manufacturing and delivery using SPCD. The SPCD shows cause and result of the accident by the propagation of exceptions. The “procurement delay” exception causes “unmanufactured orders” in “Accept order” process and “delay in production” in “Manufacture cake” process. The “delay in production” exception propagates to “insufficient refrigeration period” exception in “Refrigerate cake” process. Finally, “crumbled cake” exception has occurred in “deliver cake” process because of insufficiently refrigerated cake.

To prevent this event, it is needed to know the “procurement delay” exception in the course of “accept order” process and suspend or stop orders that will cause unexpected troubles. In this way, SPCD will help analyze exception propagation and prevent unexpected matters in business processes.

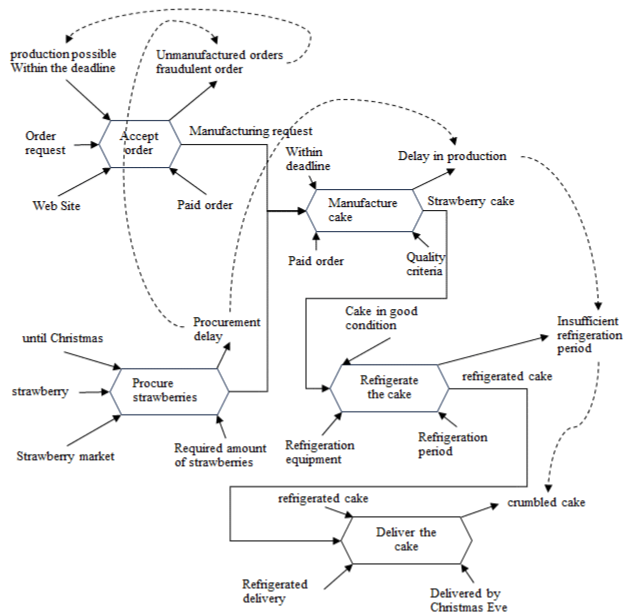

Figure 5. Example of Strawberry cake shipping

V. DISCUSSION

A. Novelty

The SPCD is designed to clarify comprehensive business process completeness by using six aspects. They are input, output and four conditions (acceptance, resource, judgement, and exception). So far, the aspect combination proposed in the paper has never been known. Moreover, exception propagation relationship has been proposed to countermeasure the failure risk of business processes. Acceptance conditions can block further failures by recognizing that an exception has occurred during the course of subsequent processing.

The completeness of business processes has also been defined by using SPCD aspects. Until now, the completeness of business processes has not been clear.

JKK needs to describe not only business process diagrams but also requirements organization sheets for processes. SPCD compactly describe comprehensive business process conditions than JKK in one diagram.

B. Effectiveness

In this paper, we proposed SPCD as a method of analyzing the completeness of business processes. In addition, we clarified the effectiveness of the proposed method by applying it to the simple service delivery and manufacturing examples. It was also revealed that the completeness of business processes can be confirmed by propagating exceptions.

C. Equivalence of SCPD

SCPD is defined by a set of processes P, aspects A, I, R, J, O, E, and relationships set R between elements of P. let P= {(x, a, i, r, j, o, e): x is a process, a, i, r, j, o, and e are aspects of x}. Then R can be the union of the following three set.

Output to Input {(x. o, y. i): x and y are processes of P}

Exception to Exception {(x. e, y. e): x and y are processes of P}

Exception to Acceptance {(x. e, y. a): x and y are processes of P}

Now, let <P1, R1> and <P2, R2> be two SPCDs.

<P1, R1> and <P2, R2> are equivalent if the following condition holds.

P1 = P2 and R1 =R2

D. Comparison FRAM and SPCD

FRAM and SPCD have common aspect as input, output, and resource. FRAM has time, precondition, and control aspects which are not in SPCD. SPCD also has acceptance, exception and judgement condition aspects which are not in FRAM. The output of FRAM is restricted to output aspects. Therefore, the meaning of output in FRAM may be unclear as it is difficult to discriminate exceptional output from normal output by aspects.

Although there are differences between FRAM and SPCD, it is unclear whether they have the same expressive power. As FRAM can be applied to analyze the resonance relationship between processes, the completeness of business processes may also be possible to analyze by FRAM. Sujan, and Felici [19] combines Failure Mode and FRAM. This implies a new method possibility that integrates analysis method using SPCD with Failure mode analysis.

The formal comparison between FRAM and SPCD is an interesting future research theme.

E. Comparison with IDEF0

The comparison of SPCD and Integrated DEFinition 0 (IDEF0) [20] is as follows. IDEF0 describes connectivity of functions with four arrows of input, output, control condition and mechanism conditions. Only output arrow of IDEF0 flows into outside functions from the source function.

SPCD describes six arrows of input, output, acceptance condition, judgement condition, resources condition and exception condition. Output and exception condition arrows of SPCD flow outside from processes.

In IDEF0, it may complicate to distinguish exception flows from output flows. Moreover, acceptance and judgement conditions are difficult to distinguish in control conditions of IDEF0.

F. Digital Transformation

The data driven management is a vision of Digital Transformation. Digital business processes will ease to collect business data in real time. The six aspects of proposed SPCD are business data candidate shall be collected for the digital twin of organizations. For example, a major business process failure incident will not be managed if management is unaware that an exception has occurred. This will be the case which mentioned episode in the beginning of this paper. The incident response should rapidly be executed. Digitalization of incident management is inevitable, because human communication is time consuming task. Moreover, human employees tend to hide incidents where they are cause or responsible. Digital algorithms do not hide incidents and, if implemented, report them quickly. This shows the importance of identifying aspects of SPCD. If the aspects are not identified, business process data cannot be collected and utilized.

G. Limitations

In this paper, we proposed a method to describe complete business processes by SPCD. We also clarified that SPCD can express the exception handling knowledge in comprehensive business processes. These cases are only based on small cases happened in Japan.

Future work on evaluating the proposed method can be designed an experiment to compare SPCD with JKK, BPM, and IDEF0. For the given same business process, it is needed to compare productivity and quality of these approaches. Moreover, qualitative capability assessment study of these approaches should be conducted.

Although the necessity of digital twin of business organizations was mentioned in the former section using SPCD, the digital twin architecture has not been clarified. The digital twin of SPCD will provide exception events monitoring and activation of appropriate handling processes. It also stores all management data issued across business processes required for data-driven business management.

VI. CONCLUSION

In this paper, we proposed a notion of business process completeness, as well as Self-Process Complete Diagram (SPCD) for describing business processes in industry. As a result, we clarified the following.

(1) SPCD can represent the business process using six aspects

(2) SPCD can represent the defect propagation process

(3) It was also pointed that SPCD has the potential to integrate business process design and data driven management of industry.

ACKNOWLEDGMENT

The author would like to thank Shinichi Sasaki, former vice president of Toyota Motor Corporation, who conceived the JKK concept through Toyota Motor Corporation's manufacturing process. The author also thanks Ms. Hosomi who introduced the JKK to the author. Without her suggestions I would never know JKK.

REFERENCES

[1] The Japan News, Irregularities result from closed corporate culture. Available from: https://japannews.yomiuri.co.jp/editorial/yomiuri-editorial/20220804-49330/

[2] S. Sasaki, "Self-process completion - Quality is built in the process," JSQC selection, Japan Society for Quality Control, 2014 (in Japanese).

[3] J. Salvadorinhoa and L. Teixeira, "Organizational knowledge in the I4.0 using BPMN: a case study," CENTERIS 2021, Procedia Computer Science 181, 2021, pp. 981–988.

[4] A. Ore, O. Kuznecova, and A. Jegorova, "Self-managed Organization: A Role of Business Process Management," 2021 62nd International Scientific Conference on Information Technology and Management Science of Riga Technical University (ITMS), 2021, doi: 10.1109/ITMS52826.2021.9615260

[5] P. Faraboschi, E. Frachetenberg, P. Laplante, D. Milojicic, and R. Saracco, "Digital Transformation: Lights and Shadows," IEEE Computer, 2023, pp. 123-130.

[6] S. Yamamoto, "A Strategic Map for Digital Transformation," KES 2020, Procedia Computer Science, vol. 176, 2020, pp. 1374-1381.

[7] S. Yamamoto, "A Comparative Analysis of Business Model Notations," Journal of Business Theory and Practice, ISSN 2372-9759 (Print) ISSN 2329-2644 (Online), vol. 7, no. 3, 2019.

[8] D. Leonard and W. Swap, "Deep Smarts: How to Cultivate and Transfer Enduring Business Wisdom," Harvard Business Review Press, 2005.

[9] Z. Nakamura, "How to conceive a simple job—Succeed in Entity Process Analysis," Nikkan Kogyo Shimbun, 2003 (in Japanese).

[10] S. Yamamoto and H. Fujimoto, "PKC-- Production Knowledge Chart for Knowledge Transfer," KES2023, Procedia Computer Science 225, 2023, pp. 414-423.

[11] D. Dori, "Model-Based Systems Engineering with OPM and SysML," Springer, 2016.

[12] B. Cameron, E. Crawly, and D. Selva, "System Architecture," PEARSON, 2015.

[13] S. Bussmann and K. Schild, "Self-Organizing Manufacturing Control: An Industrial Application of Agent Technology," 4th Int. Conf. on Multi-Agent Systems, 2000, pp. 1-8.

[14] I. Graessler, J. Hentze, and A. Poehler, "Self-organizing production systems: Implications for product design," Procedia CIRP 79, 2019, pp. 546-550.

[15] E. Hollnagel, "FRAM - the Functional Resonance Analysis Method: Modelling Complex Socio-Technical Systems." CRC Press, 2012.

[16] J. Lundberg and R. Woltjer, "The Resilience Analysis Matrix (RAM): Visualizing functional dependencies in complex socio-technical systems," In: Proceedings of the 5th Resilience Engineering Association Symposium, 2013.

[17] R. Patriarca, G. Pinto, G. Di Gravio, and F. Costantino, "FRAM for Systemic Accident Analysis: A Matrix Representation of Functional Resonance," International Journal of Reliability Quality and Safety Engineering, vol. 25, no. 1, 2018, doi: 10.1142/S0218539318500018

[18] S. Yamamoto, "Comparative Study on Functional Resonance Matrices," Knowledge-Based Software Engineering: 2022: Proceedings of the 14th International Joint Conference on Knowledge-Based Software Engineering (JCKBSE 2022), M. Virvou, T. Saruwatari, L. Jain *Editors*, Learning and Analytics in Intelligent Systems 30, Springer, 2023, pp. 169-180, doi:org/10.1007/978-3-031-17583-1

[19] M. Sujan and M. Felici, "Combining Failure Mode and Functional Resonance Analyses in Healthcare Settings," SAFECOMP, LNCS 7612, 2012, pp.364-375.

[20] IEEE, "Standard for Functional Modeling Language - Syntax and Semantics for IDEF0," IEEE Std. 1320.1-1998.