Integration of Mechanical-Electric-Software Design and Architectural Analysis: Case Study of Japanese Firms

YoungWon Park[1], Kin'ya Tamaki[2]
[1]Graduate School of Humanities and Social Sciences, Faculty of Economics, Saitama University, Saitama, Japan
[2]Department of Business, Aoyama Gakuin University, Japan

Abstract--**This paper examines the source of complexity involving synergistic products that require mechanical-electric-software interfaces integration and suggests plausible reasons of why collaboration in engineering chain (i.e., mechanical-electric-software network) is difficult to achieve. Furthermore, we analyze the integration processes of engineering and supply chain by using architectural analysis and present product development strategy responding to complexity.**

I. INTRODUCTION

With the increasing degrees of globalization and information intensive nature of work, firms wrestle with diverse and complex customer requirements. It is all the more challenging for firms to design and develop products in short product cycle time. Especially, the products that are controlled by integrative sets of numerous mechanical component parts, diverse electric circuits and sophisticated software, the level of complexity in new product development processes is accelerating with multiplying effects of the interactive functions and number of related structures plus different types of design logics for mechanical-electric- software integration requirements [12].

However, as the level of complexity of high technology products (e.g., automobile, digital instruments and high precision machineries) accelerates and the development network chain (i.e., mechanical-electric-software interfaces) is not well-coordinated, there often occurs engineering or supply chain disruption. Thus, key performance outcomes measures such as new product development productivity, lead time, design quality and entire supply chain effectiveness are all negatively affected.

Such lack of collaborative interactions are associated with organizational system differences or/and heterogeneous product development processes among mechanical-electric-software network chain. In addition, design philosophy and nature of design work are vastly different in that electric and software design focuses on functional design while mechanical design emphasizes structural design. However, collaborative design for mechanical-electric-software network requires integration of diverse cultural patterns, philosophical preferences and program languages. A key for resolving complex communication challenges might be possible through design work innovation using effective ICT system implementation.

This article examines the source of complexity involving synergistic products that require mechanical-electric-software integration and analyzes the disruption processes of engineering and supply chain and presents the responsive mechanisms. Furthermore, we suggest plausible reasons of why collaboration in engineering chain (i.e., mechanical-electric-software network) is difficult to achieve. We also provide IT system-enabled collaborative design solution for mechanical-electric-software chain through architecture analysis which we have applied to Japanese firms. The first case study discusses the engineering chain failures and subsequent corrective measures of Tanaka, a noted Japanese automotive product development firm. The second case of Yamada, a system vendor, reports the problems related to collaborative design problems of mechanical-electric-software network and the innovative solution measures.

II. LITERATURE REVIEW

A. Differences in product design ideas

As consumers' demands have become increasingly uncertain, diversified, and sophi-sticated, current products in advanced nations tend to become more complex [12]. On the other hand, to absorb such uncertainty the development periods for these products need to be reduced. Thus, it has become one of toughest challenges for today's corporations to design and develop such complex products in a short period. According to product architecture, an increase in product functionality requested by customers, quantity of structural elements such as parts corresponding to these functions, and number of correlations between the functional and structural elements of product designs lead to an increment in the number of coordinative routines and procedures required for development [1][3][4][5] [7][8][9][10][11][12][16][17][18][19][20][21][24][27][30][31]. Consequently, both corresponding products and their design processes have become more complex [25][26][28].

In electric products design, once product design concept is defined at the frontend product concept stage, external image of Printed Circuit Board (PCB) is determined. From this point forward, mechanical and electric designers divide their responsibilities and start detail design work. Concurrent to such detail design work progresses, software functionality design also starts. It is essential for electric and software designers collaborate and share design information for moving forward the product development process.

Recently, more firms introduce premium value products that have standardized control features, and yet mechanical-electric-software interfaces are not necessarily smooth [12]. Differences among designers from multiple functions present real challenge. Electric and software designers take functional specific approaches because mechanical design is more or less about controlled response and electric and software design about control initiation. As start line is different, it is quite cumbersome to synchronize the motivation aspects of each functional area. In case of design changes there might be considerable confusion.

More issues become quite problematic as the level of high product functionality requirements accelerates. With rising product complexity, the role of electric control and software real time control increases and development time is shortened. Naturally, product development structure moves toward integral architecture in greater depth and the level of perplexing confusion among the product development participants is quite common.

However, the real issue is not about start line. Communication breakdown among mechanical, electric, and software designers is the bigger issue. For example, any design change needs to refer back to the product concept or functional design stage for any problem resolution.

As any design change process moves further into the upstream, the subsequent iterative process requires involvement of the corresponding downstream. It is essential to ensure connectivity of mechanical-electric-software interfaces. Thus, the upstream planning work (e.g., concept definition and functional design) needs to consider motivational effect on the downstream implementation details. In view of diverse design approaches and communication patterns, it is crucial to install control mechanism for the entire product design and development processes.

B. Integration challenges of mechanical, electric, software requirements

1. Mechanical and electric design areas

Mechanical design and electric design work organizations are independent from each other and their work areas are usually separated. Integration of mechanical design and electric design, therefore, should consider close interactions in the form of organizational familiarity and work area proximity [2][10][13][29]). In the case of Personal Computer (PC) production, although mechanical design and electric design are separately done, open conference calls or meetings are encouraged for joint design decisions. Instead of virtual network connection the work areas of mechanical and electric engineers are closely configured. Senior level engineers (both mechanical and electric) coordinate the detailed work among the next level of engineers. Through such interactive processes, shared vision is formed in the upper level [14][22][23] and information sharing among the working level allows problem resolution in timely and cost-effective manner [15][6]. From the mechanical-electric design information sharing perspective, many firms still take analogical methods of communication and with the adoption of 3D CAD system, the layout take challenges have considerably improved. PC maker upgraded design information sharing through 3D models rather than traditional drawings on paper.

In regard to mechanical and electric design integration, another challenge is the information integration with suppliers [32]. 100% internal product development is practically impossible to attain and firms rely on their suppliers for new product development. For electronics firms that use electronic component parts, it is especially difficult to achieve supplier integration [29]. For example, notebook PC's mechanical-electric design hardly requires printed circuit board (PCB). Design of these products uses additive design methods based on combining component parts in sequence. Electornic component parts suppliers hardly submit 3D design models and instead offer image design drawings only. OEM makers themselves prepare 3D design models. PC maker also translates on its own 3D design models of component parts from the suppliers.

On the other hand, 3D virtual layout integration of mechanical- and electric design without organizational location proximity requires the huge data storage capacity and therefore 3 D viewers is commonly used instead. In that case, time lag and data confusion might still occur [29].

Any products that have thick size requirements (e.g., personal computer) may not require integration of mechanical design and electric design. However, recent smart phones that involve very thin chassis design require integration and test of mechanical and electric3D drawing. Such light, small and thin products frequently report product failures with the issues related to EMC (electromagnetic compatibility) and heat requirement. Therefore, by using a variety of simulation tests of electronic components any potential problems should be corrected in advance. After due simulation tests, the final tests are conducted with real products. In particular, EMC and heat solution are somewhat contradictory to each other. To avoid heat accumulation, holes in the product surface case might be recommended and subsequently EMC issues arise. Therefore, mutual crossover design may be a better option to respond to electric EMC and mechanical heat occurrence.

2. Integration of hard design (mechanical electric) and software design

Integration of mechanical and electric design for the same hardware products is quite feasible [12]. However, software design concept is quite different from hardware design approach and thus its management is not so straightforward. In general, software development requires extra time and cost additions for precise documentation compliance and connectivity needs among different process modules. Software design can move forward only after programming patterns are determined. In contrast, hardware design (e.g., mechanical design) shows visible product features and design process is usually fairly smooth. These differences make software design and mechanical-electric design is not necessarily compatible.

Integration of software and hardware design involves information management of customer requirements which are detailed in the form of BOM (Bill of materials) (i.e., a list of the raw materials, sub-assemblies, intermediate assemblies, sub-components, parts and their quantities). By functional units, software and hardware design updates would facilitate product development progress. Recently, general consensus is that it is not engineering design BOM but functional disclosure BOM is what manages design information [12][29].

TABLE 1. INTEGRATION OF PREVIOUS RESEARCH AND DISCUSSION

Topics	Research	Discussion
Differences in product design ideas	Clark and Fujimoto, 1991; Pine Ⅱ , 1993; Kogut and Bowman, 1995; Ulrich, 1995; Ulrich and Eppinger, 1995; Sanchez and Mohoney, 1996; Fine, 1998; Aoshima and Takeishi, 2001; Chesbrough and Kusunoki, 2001; Suh, 2001; Fujimoto, 2001; Fujimoto, 2003; Baldwin and Clark, 2000; Shintaku 2003, Shintaku, et al., 2004; Nishimura, 2004; Nobeoka, 2006; Oshika and Fujimoto, 2006; Chesbrough and Prencipe, 2008	Complexity problem
Integration challenges of mechanical, electric, software requirements	Araki, 2005; Hong et al., 2005; Fujitsu and Japan's Manufacturing Society, 2007; Fujimoto, 2007; Ueno et al., 2007; Rauniar et al., 2008a; Rauniar et al., 2008b; Doll et al., 2010; Fujimoto and Park, 2012; Hong et al., 2011; Youn et al., 2014	The relationships between functional and structural design elements differ depending upon the product architecture.

Development design process indicates that an inadequate level of design information is usually available at the front-end stage of design BOM. Naturally, integral design is not an option. If specific design-related troubles are detected, substantial degree of design work has already in progress beyond immediate intervention and correction. On the other hand, it is quite cumbersome and time-consuming if all the design patterns and other supportive details are documented in the form of undesirable and unreadable thick book for every new product development cycle. In design, functional BOM is the lifeline. However, current product design information management and PDM (Product Data Management) do not provide any recent software and hardware related information through functional and pattern indexes.

Automobiles require large production volumes and high priced products. Thus, modified design is becoming the standard of the main stream design work. Such products document all the design types and their process details first. However, it would delay the whole development process of certain products if entire design process documentation is mandatory. Such development design requirements may turn the final products into having too many development processes and incurring huge costs. Thus, the documentation focus is not for preparing documents for customers but empowering senior software design engineers and hardware design engineers who are actually responsible to manage the development process.

However, for audio products, multiple software engineers use web pages simultaneously. With any Smartphone (e.g., iPhone) these engineers are allowed to submit design ideas from anywhere. Therefore, integration of software design and hardware design (mechanical-electric design) utilizes synergistic combination of upstream customer requirements and functional engineering specifications.

C. Integration framework based on product architecture analysis

Various methods and diverse rules are tried and tested for integration of mechanical-electric-software design interfaces. For the purpose of this article, we focus on product architecture analysis method. Mechanical component parts and software control system collaborative development use both MILS (model in the loop simulation) in the upstream process and HILS (hardware in the loop simulation) for the post product development and thus achieve high reliability performance and quality excellence. Different from traditional V-shaped development, complex mechanical-electric-software design information requirements are all linked together and use two different approaches. First, one dimensional simulation examines mechanical-electric-software related system movement. Second, product requirements and mechanical-electric-software interfaces allow cost evaluation and expand integral design scope.

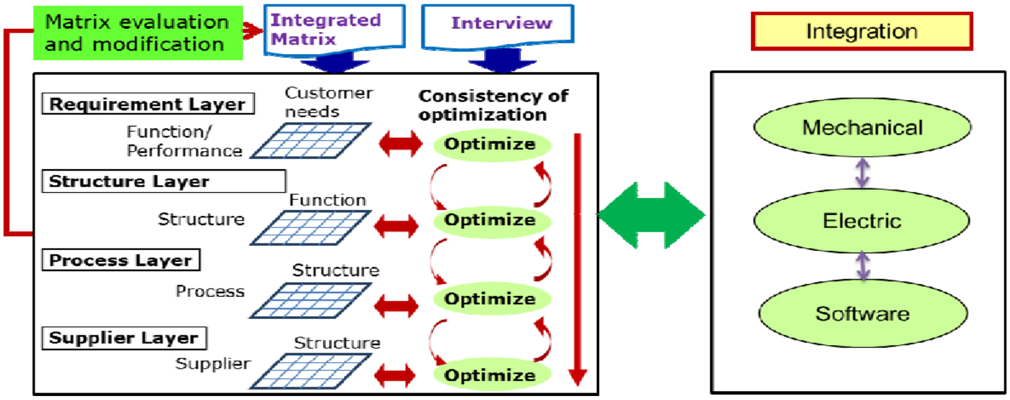

Figure 1. Research Framework

Figure 1 shows current functional requirements of mechanical-electric-software system development. Integrated matrix consists of customer needs, function specifications, process structure and supplier structure. Matrix evaluation and modification on requirement layers related to integrated matrix that are based on system development methods, software program rule logic and development work process. The extent of correlations among functionality elements, technology patterns, hardware (mechanical electric) design requirements, software development items restrictions (e.g., priority, influence and importance factors) are formally measures and thus, entire design process motivation for related product system development is possible. In this way, design process integration in the course of mechanical-electric-software collaboration facilitates real time interactive problem resolutions and rework minimization. Such architecture analysis provides incentives for the relational interactions among system requirements, diverse functionality (patterns) and software control and naturally become more responsive any possible collapse of engineering chain because of increasing productivity development complexity. Specific products that require high degree of integrality are placed at the upper architecture matrices as core design elements. Such visibility helps to reduce product development lead time and develop the organizational core competence as well. In brief, Figure 1 is a system development framework of integration of mechanical-electric-software design for any complex product development.

III. CASE STUDY

The key issue is a difficult collaboration among mechanical-electric-software engineers with different design philosophy and approach. We present a practical framework based on architectural analysis methods. In this section we now introduce the case example of Japanese firms that have applied this framework. This case specifically reports on (1) project experience details of Tanaka Co., an automotive parts development firm, in terms of engineering chain failure and subsequent response; (2) issues related to collaboration and solution hints through the dual roles (i.e., both a system vendor and system user) of Yamada Inc. in terms of responsive strategy and development practices. We organized these case studies from 2012 to September 2014 by involving senior executives and product development managers and evaluating their project management results. For field-based front end project analysis, we conducted engineering supply chain analysis and applied architecture analysis methods. The project with Tanaka consists of three senior managers and two design engineers related to product development. Additionally authors assisted to apply architecture method to design engineering development. For the project of Yamada, there were 11 members to build this architecture analysis method through discussion of authors.

A. Case of engineering chain integration failures in automotive product development

1. Collapsing damages in drive train components

It is often said that today's automotive products require the enormous configuration of electronic component parts. In particular, as the portion of electric control increases in the form of ECU (embedded software); automobile development processes are becoming more complex in contrast to the previous generations of mechanical processes. In fact, the entire automotive development system is turning to be more invisible. However, as firms fail to control these invisible process portions, often big troubles occur. In this paper, we examine Tanaka Co's (Note: for the confidentiality requirement, we refer this automaker as Tanaka Co. in this article) engineering chain collapse case occurred during automobile development process. Specifically, we introduce troubles in powertrain component parts and subsequent corrective responses.

Additional troubles occurred twice in other lines on different time periods. The break control failures had most serious impact on mechanical component parts. The first trouble was after the completion of mechanical regular preparation and finalization of mass production model blue print. In the course of test driving of the prototype automobile on the heavy snow-covered bumpy roads, drive component parts were damaged. The first trouble occurred in the front-wheel drive shaft joint that delivers power to engine. The supplier B's breaking control system caused engine power to be directed to particular wheels. Breaking control performance has rapidly improved over the past ten years. Anti-lock braking system (ABS) is for improved vehicle control and decreases stopping distances on dry and slippery surfaces for many drivers and traction control system (TCS) is designed to prevent loss of traction of driven road wheels. In the past, a simple control measure was to push brake pedal and to stop all four wheels. Independent control of four wheels allow to each wheel to respond to control signal according to road condition. The supplier B has developed its braking system in keeping up with such technological advances. B's breaking system automatically stops any wheel that is losing road grip and instead allows its outer and inner wheels to different rotation speeds and thus facilitates steering control and stability of vehicle. For example, if a car is losing control in the sands and deep snow, such braking system demonstrates its effectiveness.

2. Misplaced control design information

The fact that this innovative break system in the new car was not communicated to Tanaka's mechanical design team of power train development. Since break control aspect had not made any significant contribution to drive train development (i.e., power train components excluding engine and hereafter DTD), mechanical design team did not expect any new break system innovation. Tanaka's DTD has two processes: engineering design development phase and mass production implementation phase. Both two phases requires several times of prototype confirmation. Prior to mass production implementation phase, basic performance tests on the prototype car (e.g., strength and endurance measures) are completed in the engineering design development phase.

However, the prototype cars of any development project with initial troubles were not equipped with new break system. At the mass production stage, component parts failure occurred in a prototype test. In response, the project team decided to increase the joint part size and changed the break control logic. The team concluded that the trouble was resolved but the second trouble recurred just before start of production (SOP). This time the front part of final drive shaft failed. The root cause analysis confirmed the design change in break control system—in particular related to the system bug due to change of specific constant to break control pressure. Thus, the team had to revise the constant. Such problem arose that mechanical designers and control designer do not work under shared design rules.

3. What has kept the sharing information between mechanical and control design team?

Crucial information was not properly shared for several reasons. First, lack of routine communication between drive train development (DTD) and break control team. For example, at the first failure in front front-wheel drive shaft join, DTD started investigating the causes through various tests and learned that new control measures were added without their knowledge. At this stage DTD communicated with break control team and discussed the possible solution. Second, mismatch between mechanical and control process. Control process development in general starts in full scale immediately after the regular mechanical preparation. Control process team insists that until mechanical details are confirmed, there is no real need to hurry up the break control system development. As a result, such mismatch resulted in potential trouble spot between these two development processes. Third, another reason was that the firm counted on the break system development through its suppliers alone. All these troubles suggest that mechanism team processes too often dominated the entire automotive development processes. In concept design stage the major concerns are sales value based on mechanical aspects (e.g., choosing right type of engine or suspension system). Thus, development teams in general perceive that control aspects are considered as secondary and rarely do they discuss control aspects in the product concept stage. Without any due communication between mechanical and control processes, they finally recognized the vital inter-relationships after experiencing series of troubles.

4. Responses to invisible control requirements

Coordination challenge between mechanical and control process is mostly due to the fact that control process is invisible in nature. The solution is to visualize and share the problem issues. Tanaka firm uses certain process mechanisms to share the information related to control design model in early design stage and also receives detail design information of brake control models from its suppliers even in the form of black box. Tanaka utilizes the interpretation methods and tries to analyze the dynamic brake control change and its impact based on the information gathered from its suppliers and 3D models within. Furthermore, mechanical design engineers use 1D simulation (e.g., MATLAB/Simulink) to examine the entire system on functionality basis. The objective is to visualize the invisible control process and see the entire system through 1D simulation results and thus standardize the languages between mechanical and control design engineers. In practice, these engineers use computers installed in the car and check the control features in advance by HILS (Hardware-In-the-Loop-Simulation) methods. Control information changes up to the mass production stage, it is not feasible to check it with actual car. By using HILS, the firm may detect possible solutions to any design and then conduct actual check of the car. Another problem related to control is lack of system engineers that detect the issues related to break control. Preventive measures on the actual failures are possible but unknown break control failures are beyond remedy.

5. Integration of mechanical-electric-software design by architectural analysis

Tanaka firm implements oversight prevention measures for mechanical design engineers to check the interrelationships based on the pre-classified dependence relationships matrices between design patterns and product element structures. As shown in Figure 2, it visualizes product architecture in terms of patterns-functions-structures. In the past, Tanaka firm used QFD (quality function deployment) but data in the product concept design stage alone is so enormous that the dilemma was often expressed as, “Once data collected data no longer useful”. Their challenge was how to make productive use of data gathered through development process. Thus, they started QFD matrices (Row : functional requirements, Column: specific design features) of the entire product concept planning. Specifically, they divided design details into three decision criteria by design review process such as: (1) front end power train concept examination (e.g., determine whether use current engine or reuse of transmission), (2) power train concept examination (e.g., estimate the development budget and investment requirements and examine basic design features), (3) concept approval (e.g., cost, volume, investment amount and profit estimations).

Practical division methods include the following details. For example, for estimating engine emission amount, the final decision is about 4 cylinder and two litter engine, Then, the temporary decision item is about the engine torque as 180－210 N zone. By dividing matrices, large scale QFD shows the relationships between functions and design features. Design details are further divided into each process and by phase and data divisions are much more manageable

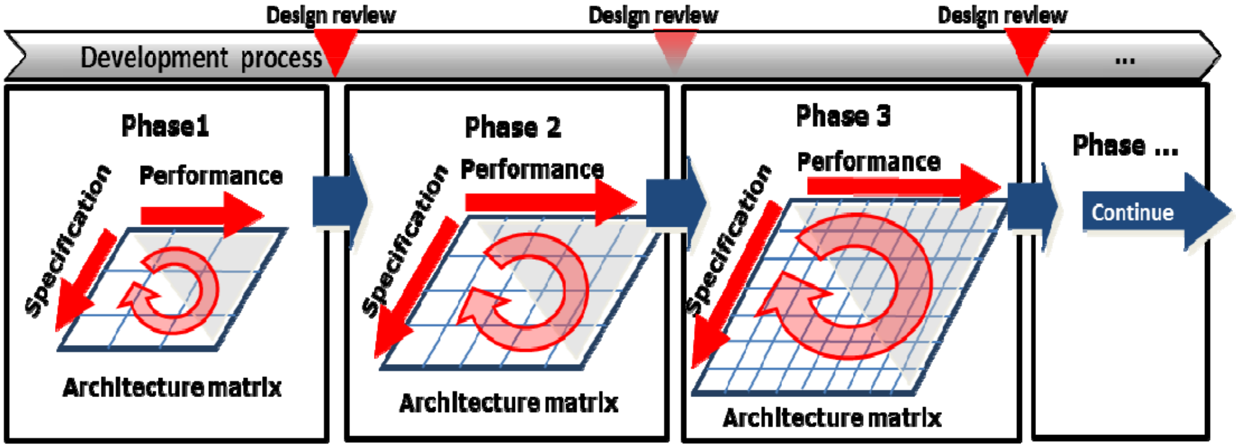

Figure 2. Architecture Analysis of TANAKA firm

than before. These matrices are then sorted by the extent of impact. In this way, not all decisions are to be made upfront, careful sorting prevents rework and instead, the specific decisions are further carefully examined as the design process advances. By applying architectural analysis, the relationships between mechanical design and control process is better understood. Accordingly, any coordinative needs are handled on timely basis. Initially Tanaka firm experienced engineering chain disruptions twice. They were due to difficulties in integrating mechanical-electric-software design with the increasing complexity of mechanical design and control process interfaces. However, with architectural analysis, this firm achieved integration of mechanical-electric-software design and therefore such troubles are now predictable and therefore manage them in advance.

Finally, such implementation of architectural analysis is based on the innovation of development organizational structure. Tanaka firm has moved away from function-based organization (e.g., propeller shaft, final drive) to system-unit based organization (e.g., A platform drive system development). For example, size minimization required system design enlargement based on system optimization goal because function-based optimum development has reached its limit for any further improvement.

Until now, the firm developed the necessary engineering designers on small scale within the project teams. Increasingly, Tanaka firm recognized the need to strategically develop groups of system engineers who are capable to integrate diverse functional requirements. Architectural analysis allows Tanaka firm to outsource any processes with low value added potentials and focus on developing key core competencies in the form of its own unique technological and human resource capabilities. In this way, architectural analysis is quite useful in assisting Tanaka’s strategic resource allocation decisions as well.

B. Responsive Actions through IT system

1. Responsive management on design changes

Yamada-firm developed VPS (virtual product simulator) for design change solution. With VPS, it is possible to check any system trouble on computer video screen and resolve any front loading-related problems. VPS allows the virtual movement of 3D data enables multiple simulations including intervention check and ensures high quality of prototype examinations. Since all the modules that move mechanical firm software is prepared, any potential trouble with control system can be examined in advance on VPS. Responsive CAD has also a wide variety of Pro/Engineers and ICAD/MX, Solid works and Inventor. Anyone who does not use 3D CAD also can use to examine the relevant data and anyone related to design and production can assess the product features as needed. Yamada-firm uses knowledge share modules across functions to share all the relevant information related to BOM, technical documents, past failure cases, and know-how concerning detection and awareness of particular product issues. Knowledge share informs all the background information behind 3 D data as well. By using key words of properties of 3D data design in the entire documents system, anyone may see the search results soon. In this way, knowledge share makes complex monozukuri information available to any interested design engineers. This enables particular design engineers not to repeat the same tasks and instead get access to past information with little difficulty. Utilization of accumulated know-how is not limited to specific products. For example, component parts information for notebook products is useful for smart phone design work as well. For smart phone design, it is possible to apply the similar product components used in the notebook PC. In such case, any trouble related information of the component parts of notebook PC is still useful to smart phone design work. Such is the strength of ICT (information and communication technologies) applicable to information sharing beyond time and space.

2. Responses through architecture analysis methods.

Besides VPS (virtual product simulator), Yamade-firm also pursues knowledge sharing through architectural analysis. At present, it is in the testing stage within the firm. For commercialization purpose, necessary system is under development.

Figure 3 shows the architectural analysis framework that Yamada firm uses. This is to achieve maximum effect to the users by considering diverse usage and application options in the form of mechanical-electric-software design. Specifically, Yamada firm strives to horizontally integrate architectural analysis, machine learning aspect of information and existing design information (e.g., BOM). At present, Yamada firm is in process of connecting architectural analysis to BOM system in the form of supporting concept design work and still needs to overcome a barrier in the formation of uniform language. Architectural element and functionality definitions are done in word. Yet, even the identical movement is expressed somewhat differently by mechanical-electric-software design engineers. For example, simple phrase like "how to move?" does contain slightly different nuances for dissimilar functions. Languages are living organisms that are subject to constant changes and therefore machine like uniformity that is applicable across functions is in need. Yamada-firm regards it quite feasible to apply current architectural analysis to improve products similar to the previous developed. Architectural analysis may derive new awareness out of embedded knowledge of accumulated system design. It therefore facilitates higher level of product planning possibilities. Thus, architectural analysis assists senior management for strategic product decisions.

Connecting the upstream engineering chain (e.g., design integration) with downstream supply chain is an important strategic concern. Architectural analysis in essence is the practical tool to integrate design information flows and therefore it is useful to integrate engineering chain and supply chain based on customer need-based information. Yamade firm management is convinced that such architectural analysis is an effective mechanism to integrate engineering chain and supply chain in the context of increasing product complexity and thus proactively respond to potential supply chain disruptions.

IV. CONCLUSION

As customer requirements are uncertain, diverse and elaborate, today's products become more complex than before. The level of complexity in mechanical-electric-software interfaces is becoming more obvious as electric control by software system over numerous mechanical component parts intensifies. Even the minimum level of knowledge and skill requirements are far beyond the range that any individual can handle because complex product development processes, technology, customer requirements, and organization management are all closely interconnected with no reasonable boundaries. Thus, additional training and IT system support are essential to enhance the engineering design capabilities. In this sense, establishing rules for knowledge management (KM) is crucial to allow individuals to master essential knowledge components. Even super engineers (or product architects) need knowledge management system that translates the embedded experiences and insights into explicit and transferrable knowledge.

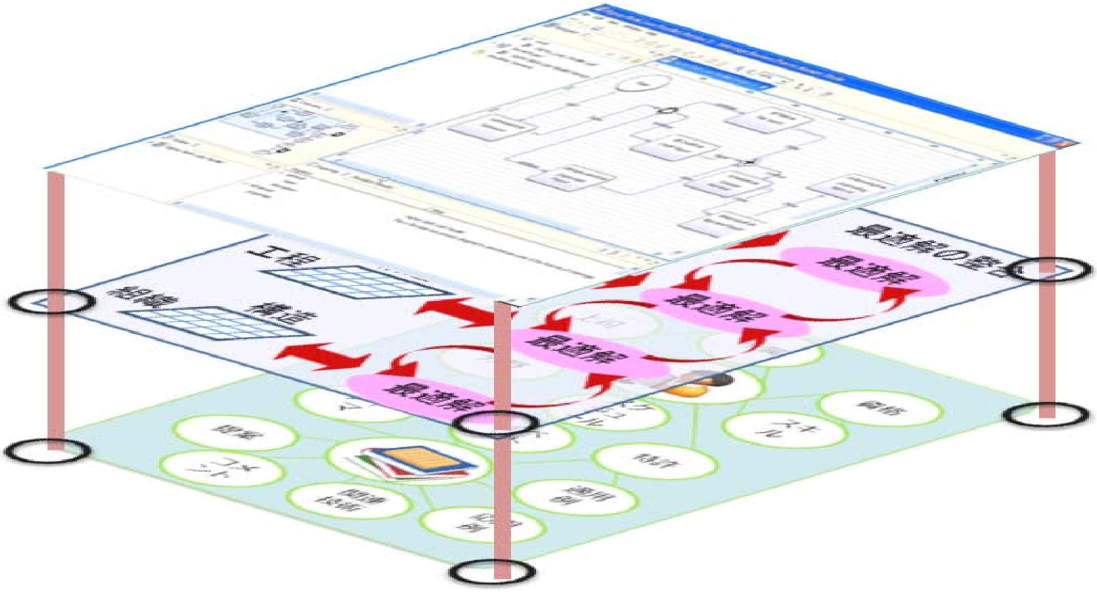

Figure 3. Architecture Analysis of YAMADA firm

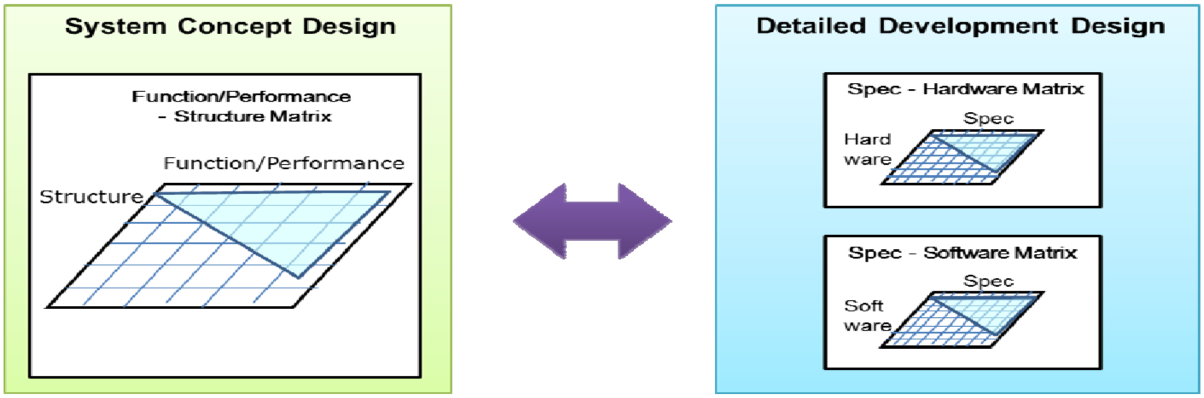

Figure 4. Integration of Mechanical/electric/Software by Architecture Analysis Method

However, such system construction itself does not guarantee trouble-free perfection. Merely accumulating embedded knowledge into massive explicit information does not necessarily increase the design engineers' problem solving skills. In fact, prudent classifications into usable information in time of need are crucial for system effectiveness. Architectural analysis would be useful as the theoretical foundation of such system design. Architectural analysis by nature classifies any embedded knowledge into "element-functionality-structure" layers and saves the data into proper matrices according to integral or modular architecture requirements.As such information structure is properly indexed, the users may search and secure necessary information on timely basis according to architectural analysis model.

This paper, based on case studies, introduces the responsive and preventive measures for engineering chain disruption and complexity issues related to new product development. Figure 4 shows integration of mechanical-electric-software design through system concept design and detailed development design processes. As shown in Figure 4, the key is to apply architectural analysis concept and coordinate different mechanical-electric-software system requirements, diverse patterns and software control relationships. Besides, firms may develop formal training system for super engineers who are capable to manage integrative development of mechanical-electric-software design through architectural analysis. Such initiatives for new product development and organization of project teams would provide very helpful tools for engineers that wrestle with integration issues related to mechanical-electric-software design. Architectural analysis by nature is to systemize development engineers' know-how (i.e., embedded knowledge) into explicit and formal knowledge for sustainable innovation. Radical and disruptive innovation requires more than architectural analysis because there might be no relevant accumulated information. Such products demand radically different functionality and pricing structure and thus require different approaches.

Japanese manufacturing industries possess outstanding technological capabilities that are mostly concentrated in large firms. Small and medium enterprises (SMEs) and venture firms may have innovative ideas but lack in technological and engineering knowledge. Certain firms might have their own unique competitive competencies and yet be unable to integrate the required comprehensive knowledge into productive results. It is desirable to utilize virtual IT system that brings together dispersed plants, design teams, the know-how of older and even retired engineers. Hence, the realization of such system architectural analysis concept is still useful in the front-end planning purpose. For example, as discussed in the Yamada-firm case, free and speedy monozukuri space requires IT integration of engineering chain and supply chain. Tanaka-firm also uses architectural analysis for integration of diverse mechanical-electric-software design engineers.

In this respect, the architectural framework in this paper might be useful as preventive and responsive mechanism for the potential engineering and supply chain disruptions. It might further provide strategic direction for global firms in developing their core competence out of embedded knowledge of complex design information chain.

Architectural analysis discussed in this paper is based on case studies of product development. Simultaneous development of numerous new products requires integrative IT systemizations. For one particular product, excel-based data would be adequate as illustrated in Tanaka-firm case. However, integration of numerous project teams demands building integrative IT systems that classify and analyze complex architecture matrices. In the future, we plan to address IT systemization issues in the subsequent papers.

REFERENCES

[1] Aoshima, Y. and Takeishi, A.; "Concept on Architecture", In *Business Architecture*, Fujimoto, T., Takeishi, A., and Aoshima, Y. (eds.), Yuhikaku, pp. 27-70, 2001.
[2] Araki, H.; *Management of Japanese Kotodukuri*, Nikkei BP, 2005.
[3] Baldwin, C. Y. and Clark, K. B.; *Design Rules, Vol. 1: The Power of Modularity*, Cambridge: MIT Press, 2000.
[4] Chesbrough, H. and Precipe, A.; "Networks of innovation and modularity: a dynamic perspective", *International Journal of*

Technology Management, Vol. 42, No. 4, pp. 414-425, 2008.

[5] Clark, K. B. and Fujimoto, T.; *Product Development Performance: Strategy, Organization, and Management in the World Auto Industry*, Boston: Harvard Business School Press, 1991.

[6] Doll, W., Hong, P., and Nahm, A.; “Antecedents and Outcomes of Manufacturability in Integrated Product Development”, *International Journal of Operations and Production Management,* Vol. 30, No. 8, pp. 821-852, 2010. .

[7] Fine, Charles. H.; *Clockspeed: Winning Industry Control in the Age of Temporary Advantage*. Perseus Book Group, MA. 1998.

[8] Fujimoto, T.; *The evolution of Manufacturing System*, Yuhikaku, 1997.

[9] Fujimoto, T.; “Industry Argument on Architecture”, In *Business Architecture*, Fujimoto, T., Takeishi, A., and Aoshima, Y. (eds.), Yuhikaku, pp.3-26, 2001.

[10] Fujimoto, T.; *Noryoku kochiku kyoso* (Capability-building competition), Chukousinsyo, (in Japanese). English translation: Competing to be really, really good (translated by Miller, Brian), Tokyo: International House of Japan, Tokyo, 2003.

[11] Fujimoto, T.; “Complexity of Artifacts and Correspondence by Manufacturers - Design of Control System and Integration of Mechanical, Electric and Software Aspects”, *MMRC Discussion Paper*, 2007.

[12] Fujimoto, T. and Park, Y. W.; “Complexity and Control: Benchmarking of Automobiles and Electronic Products”, *Benchmarking: An International Journal*, Vol. 19, Nos.4-5, pp. 502-516, 2012.

[13] Fujitsu Japan’s Manufacturing Society, *Manufacturing without Producing any Product*, Nikkagiren Shuppannsha, 2007.

[14] Hong, P., Doll, W., Revilla., E., and Nahm, A.; “Knowledge sharing and strategic fit in integrated product development Projects: An Empirical Study”, *International Journal of Production Economics,* Vol. 132, No. 2, pp. 186-196, 2011.

[15] Hong, P., Vonderembse, M., Doll, W., Nahm, A.; “Role Changes of Design Engineers in Integrated Product Development”, *Journal of Operations Management*, Vol. 24, No. 1, pp.63-79, 2005.

[16] Kogut, B. and Bowman, E. H.; “Modularity and Permeability as Princiiples of Design”, *in Redesigning the Firm,* edited by E. Bowman and B. Kogut, New York, NY:Oxfrod University Press, pp.243-260, 1995.

[17] Chesbrough, H. W. and Kusunoki, K.; “Dynamic Shift of Product Architecture: Trap of Virtual Organization”, In *Business Architecture*, Fujimoto, T., Takeishi, A., and Aoshima, Y. (eds.), Yuhikaku, pp.263-285, 2001.

[18] Nishimura, K.; *Invisible structure change of Japanese economy*, Nihon Keizai Shinbunsya, 2004.

[19] Nobeoka, T.; *Introduction to MOT,* Nihon Keizai Shinbunsya, 2006.

[20] Oshika, T., and Fujimoto, T.; “Positive analysis of the product architecture theory and international trade theory”, *Akamon Management Review*, Vol. 5, No. 4, pp. 233-271, 2006.

[21] Pine Ⅱ, J. B.; *Mass Customization*, Cambrigde, MA: Harvard Business School, 1993.

[22] Rauniar, R., Doll, W., Rawski, G., Hong, P.; “The Role of Heavyweight Product Manager in New Product Development”, *International Journal of Operations and Production Management*, Vol. 28, No. 2, pp.130-154, 2008a.

[23] Rauniar, R., Rawski, G., Doll, W., and Hong, P.; “Shared Knowledge and Product Design Glitches in Integrated Product Development”, *International Journal of Production Economics*, Vol. 114, No. 2, pp.723-736, 2008b.

[24] Sanchez, R. and Mahoney, J. T.; “Modularity, Flexibility, and Knowledge Management in Product and Organization Design”, *Strategic Management Journal*, Vol. 17, pp. 63-76, 1996.

[25] Shintaku, J.; “Division of Labor with China from Standpoint of Architecture Theory”, *Japan Machine Export Union JMC journal*, November, pp. 2-7, 2003.

[26] Shintaku, J., Yoshimoto, T., and Kato, H.; “Strategy of Japanese firms in Chinese module type industry”, *Akamon Management Review*, Vol. 3, No. 3, pp. 95-114, 2004.

[27] Suh, N.P. (1990), *The Principles of Design*, Oxford University Press, New York.

[28] Thomke, S. and Fujimoto, T.; “The Effect of Front-Loading Problem Solving on Product Development Performance”, *The Journal of Product Innovation Management*, Vol. 17, No. 2, pp. 128-142, 2000.

[29] Ueno, Y., Fujimoto, T., and Park, Y.W.; “Complexity of artifacts and Mechanical/ Electrical Design: Mainly in CAD use of auto and electronic industry”, *MMRC Discussion Paper,* No. 179 (in Japanese), 2007.

[30] Ulrich, K. T.; “The Role of Product Architecture in the manufacturing Firm”, *Research Policy,* Vol. 24, pp. 419-440, 1995.

[31] Ulrich, K. and Eppinger, S. D.; *Product Design and Development*, McGraw-Hill, 1995.

[32] Youn, S., Yang, M., Kim, J., and Hong, P.; “Supply chain Information Capabilities and Performance Outcomes: An Empirical Study of Korean Steel Suppliers”, *International Journal of Information*, Vol. 34, No. 2, pp.369-380, 2014.