Applying Integrated R&D Process in Process Innovation Research: Estimating the Impact of a Process Change in Automotive ECU Development on Organizational Flexibility and Product Quality

Yuji MIZUKAMIa*, Hideo OWANb

aNihon University, 1-2-1 Izumicho, Narashino, Chiba 275-8575, Japan
bUniversity of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-0033, Japan

Abstract

Japanese automobile manufacturers have actively activated activities to incorporate highly intelligent function to enhance their competitiveness. As part of this policy Japanese automobile manufacturers have required the suppliers of electrical control units (ECUs) to coordinate their software development activities to improve the effectiveness and stability of in-vehicle LAN network system which should offer high intelligent function. The purpose of the paper is to evaluate the impacts of a change in the process of software development from the traditional sequential approach to the one which is more integrated and adaptive to changes in the course of optimization. We conduct regression analyses (OLS and Tobit) utilizing project management data collected from an American automotive parts supplier over the six years between January 1, 2003 and December 31, 2008. Our empirical analysis can be summarized by three key findings. First, the adoption of the new integrated R&D process, it increased the frequency of specification changes that presumably helped to improve the effectiveness and stability of in-vehicle LAN network system. Second, the new process significantly raised number of flaws caught during the development, thus improved product quality was observed after the shipment. Third, the introduction of the new process lowered productivity and raised the wage cost substantially. Additional implication for the role of firm-specific experience is also discussed.

Keywords: chain linked model, automotive, embedded software, development project, process improvement

1. Introduction

Despite the fact that Japanese automobile industry joined global competition later than American and European counterparts, it succeeded in gaining competitive advantages by the nineteen-eighties [1]. Many studies in the past thirty years have identified the constant collaboration between car manufacturers and parts suppliers as one of the most important sources of its competitive advantage [2][3][4][5][6][7][8].

Long term buyer-supplier relationship prevalent in the Japanese automobile industry has encouraged persistent collaboration in product development and quality control between car manufacturers and parts suppliers without establishing a legally binding agreement. It has been widely believed that such loose governance structure characterized by what economists call “relational contract” has facilitated the “integral architecture,” a product design architecture that relies on coordinated efforts across parts suppliers to maximize functional performance of final products [9].

However, most of the discussions emphasizing the merit of integral architecture are reminiscences of the nineteen-eighties when automotive mechanism were mostly controlled by mechanical control unit (MCU), and less than 1 percent of average intermediate input costs were spent for developing electrical control unit (ECU). Since then, information technology has revolutionized how cars are designed and manufactured. In 2005, ECUs account for roughly 15 percent, 30 percent, and nearly 50 percent of the total intermediate input costs for producing compact car, luxury car, and hybrid car, respectively [10]. Moreover, those control units are connected by in-vehicle local area network (LAN), which allows multiple ECUs to achieve more advanced functions jointly and interactively. For example, life-saving automotive air bag ECU for passenger is activated when pressure sensor ECU

*Corresponding author: mizukami.yuuji@nihon-u.ac.jp

Received: February 23, 2017
Accepted: November 21, 2017
Lessons & IHCGBEGCIDE3FDGDGBFG3ED

Some of recent research on active noise control focuses on developing an ECU for active noise control. The ECU is designed to emit a sound wave with the same amplitude but with inverted phase to the original sound. Developing such an ECU requires considerable coordination among audio suppliers in order to maximize functional performance during product development. The lack of studies hinder our understanding of many key questions regarding the effect of integral architecture on software development.

In this paper, we argue that a newly devised process of software development has improved the functional performance of the car unit level by offering many opportunities for coordination with customers and other suppliers.

Unfortunately, in contrast to the extensive theoretical and empirical literature that analyzed traditional integral architecture, there exist few studies, even descriptive case studies, that examined how integral architecture is operationalized in the context of software development. Note that integral architecture necessitates communication and coordination between car manufacturer and parts suppliers and even among parts suppliers.

For passenger seat detects passenger. Although the development of ECUs in early days required car manufacturers to pre-specify the functionality of individual ECUs like many other general purpose parts procured in the market, the development of ECUs in recent years are increasingly integrated in the development of an entire system of automobile in order to enhance the total performance of the car unit level. In this paper, we argue that a newly devised process of software development has improved the functional performance of the car unit level by offering many opportunities for coordination with customers and other suppliers.

Unfortunately, in contrast to the extensive theoretical and empirical literature that analyzed traditional integral architecture, there exist few studies, even descriptive case studies, that examined how integral architecture is operationalized in the context of software development. Note that integral architecture necessitates communication and coordination between car manufacturer and parts suppliers and even among parts suppliers.

One recent example is the coordination among audio manufacturers to develop an ECU for active noise control. The level of noise inside a car depends on the noise emission from sources such as the engine and the use of sound insulation such as noise absorbing material and dumper. Some of recent luxurious cars carry the ECU for active noise control to enhance the sound insulation performance. Active noise control ECUs are designed to emit a sound wave with the same amplitude but with inverted phase to the original sound. Developing such ECUs requires considerable coordination among audio manufacturers and other suppliers such as those of engines and car body to account for the characteristics of noise from various sources and cancel noise effectively.

Figure 1 is the Chain-Linked model for Software Development that involves substantial communication and coordination between car manufacturer and parts suppliers in order to maximize functional performance during product development. The lack of studies hinder our understanding of many key questions regarding the effect of integral architecture on software development.

This paper empirically investigates the following questions: how is close communication and coordination operationalized in software development? What is the impact of “integrated” software development on flexibility, quality, cost, delivery and productivity? How does the change toward “integrated” software development affect the role and value of human capital? The answers to the questions will have rich implications for R&D project management that involves substantial software development.

Our approach to the empirical analysis of software development project teams follows in the spirit of empirical software engineering in the sense that we use statistical analyses of data from software development to evaluate the impact of a change in software development processes.

Figure 2 is the Traditional Iterative V model that involves substantial communication and coordination between car manufacturer and parts suppliers in order to maximize functional performance during product development. The lack of studies hinder our understanding of many key questions regarding the effect of integral architecture on software development.

Our approach to the empirical analysis of software development project teams follows in the spirit of empirical software engineering in the sense that we use statistical analyses of data from software development to evaluate the impact of a change in software development processes.

Figure 3 is the Chain-Linked Iterative V model that involves substantial communication and coordination between car manufacturer and parts suppliers in order to maximize functional performance during product development. The lack of studies hinder our understanding of many key questions regarding the effect of integral architecture on software development.

Our approach to the empirical analysis of software development project teams follows in the spirit of empirical software engineering in the sense that we use statistical analyses of data from software development to evaluate the impact of a change in software development processes.

International Journal of Japan Association for Management Systems
Applying Integrated R&D Process in Process Innovation Research: Estimating the Impact of a Process Change in Automotive ECU Development on Organizational Flexibility and Product Quality

Table 1: Date of Project Formation, Type and Size

| Project | Date of Project Team Formation | Project Type | Total Phase | Term [Weeks] | Email with customer | Email with customer [All] | Email with customer [English only] | Email with other suppliers | Email with other suppliers [All] |
|---------|--------------------------------|--------------|-------------|-------------|-------------------|-------------------------|-------------------------------|--------------------------|-----------------------------|
| 1       | Jan. 5, 2003                   | TIV          | 4           | 26          | 1221              | 454                     | 266                           |                          |                             |
| 2       | June 5, 2003                   | TIV          | 4           | 22          | 675               | 153                     | 129                           |                          |                             |
| 3       | July 20, 2003                  | TIV          | 3           | 18          | 584               | 179                     | 153                           |                          |                             |
| 4       | Sept. 7, 2003                  | TIV          | 3           | 16          | 101               | 22                      | 35                            |                          |                             |
| 5       | Dec. 7, 2003                   | TIV          | 3           | 17          | 695               | 38                      | 147                           |                          |                             |
| 6       | Feb. 8, 2004                   | TIV          | 3           | 17          | 245               | 85                      | 52                            |                          |                             |
| 7       | May 23, 2004                   | TIV          | 2           | 12          | 424               | 160                     | 107                           |                          |                             |
| 8       | July 11, 2004                  | TIV          | 1           | 5           | 156               | 44                      | 42                            |                          |                             |
| 9       | Aug. 15, 2004                  | TIV          | 2           | 12          | 62                | 18                      | 23                            |                          |                             |
| 10      | Aug. 22, 2004                  | TIV          | 2           | 13          | 152               | 39                      | 51                            |                          |                             |
| 11      | Oct. 17, 2004                  | TIV          | 2           | 11          | 253               | 71                      | 63                            |                          |                             |
| 12      | Dec. 16, 2004                  | TIV          | 3           | 18          | 585               | 161                     | 140                           |                          |                             |
| 13      | Jan. 23, 2005                  | TIV          | 2           | 11          | 94                | 35                      | 17                            |                          |                             |
| 14      | Feb. 27, 2005                  | TIV          | 3           | 13          | 505               | 149                     | 135                           |                          |                             |
| 15      | June 5, 2005                   | TIV          | 2           | 17          | 535               | 180                     | 93                            |                          |                             |
| 16      | July 24, 2005                  | TIV          | 3           | 23          | 382               | 0                       | 120                           |                          |                             |
| 17      | Jan. 8, 2006                   | CLIV         | 5           | 20          | 728               | 148                     | 373                           |                          |                             |
| 18      | Mar. 26, 2006                  | CLIV         | 3           | 16          | 121               | 0                       | 60                            |                          |                             |
| 19      | May 7, 2006                    | CLIV         | 4           | 17          | 761               | 30                      | 242                           |                          |                             |
| 20      | Aug. 20, 2006                  | CLIV         | 3           | 12          | 1001              | 31                      | 285                           |                          |                             |
| 21      | Sept. 17, 2006                 | CLIV         | 2           | 15          | 252               | 0                       | 106                           |                          |                             |
| 22      | Oct. 8, 2006                   | CLIV         | 5           | 23          | 1458              | 64                      | 300                           |                          |                             |
| 23      | Nov. 12, 2006                  | CLIV         | 3           | 17          | 1318              | 58                      | 384                           |                          |                             |
| 24      | Feb. 18, 2007                  | CLIV         | 4           | 16          | 854               | 295                     | 169                           |                          |                             |
| 25      | May 6, 2007                    | CLIV         | 2           | 12          | 556               | 197                     | 180                           |                          |                             |
| 26      | May 27, 2007                   | CLIV         | 4           | 16          | 549               | 0                       | 116                           |                          |                             |
| 27      | Aug. 19, 2007                  | CLIV         | 4           | 17          | 1307              | 408                     | 470                           |                          |                             |
| 28      | Oct. 18, 2007                  | CLIV         | 2           | 9           | 589               | 206                     | 326                           |                          |                             |
| 29      | Jan. 20, 2008                  | CLIV         | 5           | 16          | 1037              | 0                       | 308                           |                          |                             |
| 30      | June 8, 2008                   | CLIV         | 4           | 21          | 752               | 0                       | 228                           |                          |                             |

Note: TIV: Traditional Iterative V model, CLIV: Chain-Linked Iterative V model

Historically, AMRD facilities in the world have applied a spiral model [24] to software development process and followed a continuous loop of steps required under AMRD software process improvement initiative as proposed by the IDEAL model of CMMI-DEV. However, Software development team in AMRD automotive parts development facility in Tokyo (AMRD-Tokyo) has adopted a V model as the de-facto standard of Japanese automotive industry.

The V model, which also provides a continuous loop of steps required under software process improvement, evolved at AMRD-Tokyo and we call it Traditional Iterative V model (TIV). Figure 2 presents a general flow chart (a) and a Gantt-chart (b) of TIV.

3 AMARD Corporation is the anonym used in the paper to conceal the identity of the company analyzed.
4 SPL (2001) is engineering techniques for creating a portfolio of similar software products from shared software assets. The traditional SPL consists of “Core”, “Production” and “Management”. The “Core” is software development of shared software assets. The “Production” is software production utilizes the shared software assets. The “Management” conducts all activates.
5 IDEAL model (1996) is a software process improvement process model comprised of five phases: initiating phase, diagnosing phase, establishing phase, acting phase and learning phase.
6 CMMI –DEV (2001) is the abbreviation of capability maturity model integration for development which is a software process improvement process guideline that comprises the IDEAL model as an integrant model.
7 V model (1997, 2005) is a development process which is the extension of the waterfall model. Instead of moving down, the process steps are bent upwards after the coding step, to form the V shape. The V model demonstrates the relationships between design steps and testing steps.

Vol. 9 No.1 (2017)
Recently at AMRD-Tokyo, TIV model further evolved into what we call Chain-Linked Iterative V mode (CLIV). CLIV is a V model derived from applying Chain-linked model for Software Development (Figure 1) to the original TIV. CLIV was invented in AMRD’s effort to adapt to integral architecture in the Japanese automobile industry. Figure 3 presents a general flow chart (a) and a Gantt-chart (b) CLIV [19].

In the embedded software development section of AMRD-Tokyo, in most cases, no more than two projects run simultaneously and all engineers participated in one of the projects or both. Therefore, the way to select project members is simply to assign engineers who become available first to the next project. However, software development engineers and software test engineers are distinguished and need to be balanced in numbers in general.

2.2. Chain-Linked model for software development

The most important concept of Kline’s Chain-Linked model is “Market Insight and Development” [20]. In the application of Chain-Linked model for Software Development, "Integration test" corresponds to “Market". The integration test aims to confirm the function of vehicle level, which is indicated by the car manufacture (customer). From the parts manufacturer’s point of view, the functional specification of the vehicle level regulated by the car manufacturer is taken as "Market", and it is important to enhance the quality of functional specifications at the vehicle level. In other words, in the process of the application of Chain-Linked model for Software Development, performing the integration test at an early stage and evaluating the performance of the vehicle function level, is more likely to stimulate the market competitiveness [19].

Figure 2 (b) and Figure 3 (b) present Gantt-charts of TIV and CLIV. It shows that the Integration Test of CLIV is being implemented at an early stage, and that knowledge is closely shared with the next phase. The scope of knowledge is consist of defects, improvement points and inconsistency of specification understanding among companies based on the result obtained by enforcement of the Integration test, furthermore know-how gained from experience is also included. The form of knowledge is primarily documented, but there are things that are shared by sharing places without being documented. In the Integration test, the car manufacturer and related companies share gathering places and perform vehicle level operation checks.

The comparison between TIV and CLIV led [19] to conclude that there is an isomorphism between product design and software development process. Namely, their architectures tend to coincide with each other in operational form. TIV adapts to modular architecture while CLIV corresponds to integrated architecture.8

CLIV employs five kinds of testing methods: automation test, procedure test, software test, unit test, and integration test (ECU network integration test). The automation test monitors how addition of new functions affects the behavior of existing functions by continuously evaluating inputs and outputs of ECU using an automatic test machine that is executed every evening. The purpose of automation test is to prevent new functions from causing any undesirable effects on existing functions. The procedure test examines the behavior of the source code one by one as the white box test. Procedure test must be executed by the developer of the source code along with another engineer when it is updated. The software test examines the behavior of the source code at the function level as the gray box test. The test must be executed before software release by other engineers who understand the software design architecture. The unit test monitors the functionality of new functions by examining whether their outputs coincide with the specifications for various inputs as the black box test. The test must be executed before software release by software test specialists in the quality assurance section. The integration test is a joint operation with other suppliers covering a network of ECUs and examines whether the outputs from the system of ECUs coincide with the specifications. In CLIV, the integration test is executed in an earlier stage of software development than in TIV and, as a result, ensures the opportunities for specification fine-tuning and sufficient time for improvement in integrated performance with isomorphs, i.e. similar in form [9], [25] and [26].

---

8 Note that a number of past works argue that the architecture of product design and that of organization structure tend to be isomorphs, i.e. similar in form [9], [25] and [26].
other ECUs.

In Automation test, Software regression test is executed. The purpose of performing this test is to confirm the influence on other functions due to the change of the software program [27]. AMRD has prepared exclusive test benches and fixture common to the organization as an automatic test machine, and the machine is designed to operate in an unattended environment at night. The test cases are programmed in Python language.

After the introduction of CLIV, communication with automobile manufacturers and other automotive parts suppliers has become more frequent and important and, therefore, non-Japanese engineers were gradually replaced by Japanese ones during our observation period.9

3. The AMARD data

This paper utilizes three types of internal data from AMRD that were given to us to evaluate the impact of CLIV: the records of software development projects, the weekly progress records for individual software development projects, and the personnel records of engineers in engineering division at AMRD-Tokyo, changed. The personnel records of employees include weekly information on worker pay and hours worked as well as basic individual characteristics such as age, tenure, and education level for all individuals employed in engineering division over this period. Table 1 reports the date of project formation, project sizes, and number of emails. There are two particular features of technology used at AMRD that are worth noting. First, the major part of ECU hardware platforms and software development methods did not change for the six years of our entire observation. It is typical of the automobile industry to use matured and proven technology that has been used for long time.

Second, required specifications for ECU kept being upgraded and increasingly became challenging over the six years. The project records contain overall information on project characteristics such as the type of V models (TIV vs. CLIV), the number of total phases, the number of requirements imposed, the list of assigned engineers, and the number of lines of developed code for all software development projects undertaken over this period. The weekly progress records consist of weekly information on the phase worked, the treated requirements, the completed lines of developed code, the hours worked, the errors discovered on integration test, unit test and software test, and the specifications. The impact of process change from TIV to CLIV can be most strikingly observed in the frequency of specification changes, which can be interpreted as flexibility in software development. Figure 4 presents the number of specification changes per 1000 lines of code after integration test for all projects in our dataset. Note that CLIV was first introduced to project team 17. The average frequency of specification changes under CLIV is six times greater than that under TIV, which is consistent with our view that CLIV was introduced to AMRD in order to have flexibility to accommodate specification change requests from car manufacturers to enhance the vehicle-level functionality in constant collaborative efforts with the manufacturers and other suppliers [19].

Figure 5 illustrates how the frequency of communication with the customer and other suppliers measured by the number of emails exchanged per week changed across projects. Communication with both the customer and suppliers almost doubled after the introduction of CLIV. It seems to suggest that augmented flexibility and the actual needs for coordination due to increased specification changes led to the increase in inter-firm

---

9 Most of non-Japanese engineers are contract workers from Asian nations, whose labor costs are lower than regular employees.
communication across organizations participating in the development of a new car model.

In the next section, we examine how CLIV affected quality, cost, delivery, and productivity using phase-level information and employing more rigorous statistical methods such as OLS and Tobit.

4. Empirical analysis

In this section, we investigate the three related sets of issues raised earlier. Table 2 presents a summary of variables for the empirical analysis, and Table 3 presents summary statistics for the project data, both overall and by project type status. First, to what extent does the adoption of CLIV increase or decrease quality, cost, delivery and productivity? When comparing module architecture and integral architecture, the latter requires more close communications with car manufacturers and other parts suppliers, and facilitates more frequent deliveries and productivity.

Second, are teams more productive as the general working experience of members is higher on average, or as the firm-specific skill is more accumulated? How did the role of firm-specific skill change after the introduction of CLIV? Thirdly, how important is close and dependable communication with customer in implementing CLIV successfully? We measure the accuracy and sufficiency of customer communication by counting the share of English emails exchanged with the customer. Our presumption is that the more communication is done in English, the lower the amount and accuracy of information communicated because subtle and hard-to-describe but potentially very important information could be lost when engineers communicate in English. Under the TIV, it was not uncommon that non-Japanese team members took the customer interface role but, as non-Japanese engineers were gradually replaced by Japanese ones under CLIV as we stated earlier, communication in English declined over time.

4.1. Impact for Quality

The following three types of defect density measures are used to evaluate the quality of teamwork: (1) unit test defect density is the number of detected errors per 1000 lines of code at the unit test; (2) integration test defect density is the number of detected errors per 1000 lines of code at the integration test; (3) market defect density is the number of errors per 1000 lines of code discovered within one year after its shipment. Table 4 presents the results of the Tobit regression analyses for three defect density measures where dependent variables take the form ln(defect density + 1). Note that this dependent variable takes zero when no errors are detected. Logarithm is taken because our preliminary observation shows that the relationship is likely to be non-linear, and Tobit is chosen because the distribution is censored at zero (i.e. a difference in quality cannot be detected among many observations where no errors were discovered). Three models are presented for each defect density measure taken as dependent variable. White's robust standard errors are used to correct for heteroscedasticity. Model 1 is a base model that includes six independent variables in addition to our focal variable, New V Model, to control for project characteristics including team size (number of team members), workload (number of required specifications), flexibility (number of specification changes to improve functionality), and capability of team members (general experience and tenure). The variable, Phase Ratio (= phase number/total number phases), is created to capture phase-specific differences in defect density. Since the total number of phases, ranging between two and five, differs across projects, we use the ratio rather than the phase dummies and include its square to account for non-linearity. Model 2 adds to Model 1 the interactive term between New V Model and Tenure (i.e. product of the two variables) to account for the possibility that the to nonnegative and quite a few observations have zero defect density [28].

![Table 3: Summary Statistics](image-url)
effects of the worker experience differ between TIV and CLIV.

Column 1, 3, 5 of Table 4 exhibit the estimation results for our base model, Model 1, that is primarily used to evaluate the impact of the adoption of CLIV on various quality measures. There are two notable results. First, the adoption of CLIV raises the number of defects detected at the integration test while it reduces those discovered after the shipment. The result is quite reasonable given that integration tests are conducted before the unit tests are executed under CLIV. More critical errors found at early stage of software development help engineers to substantially improve the entire design of software code, which in turn leads to lower market defect density. The level of defect density at the integration test is seven times greater while the level of market defect density is more than 40 percent lower after the introduction of CLIV based on Model 1. The impact of CLIV on the unit test is shown to be insignificant.

Second, tenure (i.e. software development experience at AMRD) rather than general work experience as software engineer is positively (negatively) associated with the quality (defect density) both at the integration test and after-shipment evaluation. This may imply that teamwork experience at AMRD offers tacit knowledge helpful to detect errors and such acquired skills are highly firm-specific.

Column 2, 4, 6 of Table 4 present the estimation results for Model 2, which is employed to analyze the role of human capital and study how it changed after the shift from TIV to CLIV. As the interactive terms between New V Model and Tenure show, the quality measures from both integration test and market performance have become less dependent on tenure, a proxy for firm-specific human capital, under CLIV than under TIV. This result is more clearly demonstrated in Figure 6 and Figure 7 where...
predicted defect density measures for both TIV and CLIV are drawn based on the estimated parameter values from Model 2 in Table 4. Note that defect density under CLIV does not depend much on work experience at the firm (almost independent for the integration test) while that under TIV is much more sensitive to tenure.

Why has there been such a drastic change in the role of firm-specific human capital?; Under TIV, it is critical to find as many errors as possible at unit tests for timely completion of the project. If integration tests reveal numerous errors, correcting them will delay the entire development process including those of car manufacturers and other parts suppliers. Longer teamwork experience at AMRD significantly reduces such risks. In contrast, under CLIV, integration test is executed at much earlier stage and, therefore, engineers will have sufficient time for identifying and correcting the errors. In this case, insufficient teamwork experience at AMRD is much less likely to result in delay of entire development process or affect market quality performance.

4.2. Impact for Productivity, Delivery and Cost

Since variances of both output and time required for each phase seems to vary across phases, White’s robust standard errors are used to correct for Heteroscedasticity. Models that are similar to Table 4 are presented. Model 1 is a base model that includes five independent variables to control for project type, team size (log of the number of team members), workload (number of requirements), and capability of team members (general experience and tenure).

Table 5 presents the results of the OLS and Tobit regression analyses that utilize the detailed data of productivity, delivery and cost for each phase. Column 1, 3, 5 of Table 5 offer several important results. First, the introduction of CLIV lowered productivity and raised the labor cost substantially as was concerned while it did not affect the delivery performance at all. The impacts on the first two are enormous. The point prediction based on the above estimation shows that the productivity dropped more than 30 percent on average while the cost jumped more than 50 percent. Part of the increase in labor cost can be attributed to the switch from cheap contract workers to more expensive regular employees.

There are two primary reasons behind the productivity drop. First, the time spent for communication with Japanese car manufacturers and other parts suppliers has increased. Integration test is executed in earlier stage under CLIV, and used to optimize interface specifications among ECUs in collaboration with other suppliers. Since this process requires substantial time for perfection, workload has increased significantly. Second, the number of phases has increased from the average of 2.625 under TIV to the average of 3.571 under CLIV. We believe that team members who are not used to work under CLIV became cautious and punctuated each project into smaller phases. Increasing the number of phases, however, resulted in an increase in the number of activities required at the start and end of each phase, contributing to lower productivity.

The second notable result is that larger teams are less productive and more costly. It may be because free-riding is more severe in larger teams or because more difficult
software development is assigned to larger teams.

Thirdly, general work experience seems to have a positive impact on productivity while the tenure does not have significant effect on any performance measures in Table 5. It is in contrast with our earlier finding that it is tenure, not general experience that affects the quality of the work done. Model 2 (Column 2, 4, 6 of Table 5) accounts for a possible change in the role of firm-specific human capital after the shift from TIV to CLIV. Unlike our earlier findings for defect density, we don’t see any significant change in the role of firm-specific human capital or effect of customer communication.

4.3. Effect of less informative email communication

Table 6 presents the results of the OLS and Tobit regression analyses including English_Mail_Ratio and replacing the interaction between tenure and New V Model with that between English_Mail_Ratio and New V Model in all regression models presented earlier. We repeated regressions for all six dependent variables–three-stage defect density measures, productivity, delivery and cost using the same other control variables. Note that Model 2 adds to Model 1 the interactive term between New V Model and ratio of English mail to account for the possibility that frequent and dependable email communication becomes more important after the switch from TIV to CLIV.

Column 3 of Table 6 indicates that market defect density is higher for projects where a larger share of email communication is done in English, which is consistent with our expectation. It may suggest that the supplier’s communication with automobile manufacturers became less effective presumably because problems and necessary specification changes were communicated less accurately to each other resulting in higher defects discovered after the shipment. The lower stand of Column 3 also suggests that this effect is higher under CLIV although the difference is only weakly significant.

The above result should be interpreted with caution, however, because a higher share of English emails does not seem to have affected other dimension of development performance such as productivity, delivery and other test performance. One possible explanation is that when an engineer dispatched from AMRD-US (core software development team) is better informed of all issues and communication with a customer (because most emails are written in English), the project team can receive technical support from AMRD-US more easily, which in turn might have offset the negative impact of less effective customer communication on other performance measures.

5. Conclusion and Discussion

Our empirical analysis can be summarized by four key findings. First, the adoption of the new integrated R&D process increased the frequency of specification changes that presumably helped to improve the effectiveness and stability of in-vehicle LAN network system. This increase reflects flexibility enhanced by CLIV and collaborative product improvement built in the integral architecture embraced by the Japanese automobile industry.

Second, the new process significantly raised a number of flaws caught during the development, and thus improved product quality measured by the number of defects found after the shipment. The purpose of introducing CLIV was to facilitate collaborative ECU development demanded by Japanese car manufacturers, but the new process also turned out to be very effective in improving product quality.

Third, the introduction of CLIV lowered productivity by more than 30 percent and raised the labor cost by more than 50 percent. This finding raises the question of whether the shift from TIV to CLIV is efficient or not. In other words, does the benefit of having higher flexibility and better product quality exceed the cost of bearing higher labor cost? Answering such a question requires further analysis of the firm’s financial reports and will be left for future research.

Fourth, our study reveals that CLIV has reduced the difference in quality of work between engineers with long tenure and those with short tenure by advancing workload to earlier stage of process thus leaving sufficient time to correct errors and improve the software design. This finding that firm-specific human capital may become less valuable in software development process that requires close collaboration with partner firms is counter-intuitive and requires further investigation.

Table 6: Email communication with customer
(coefficients of English_Mail_Ratio (log) in all performance regressions)

| Column | Dependent Variable | Estimation Method | Integration Test | Unit Test | Market Test | Productivity (log) | Delivery (log) | Cost (log) |
|--------|--------------------|-------------------|-----------------|-----------|-------------|--------------------|---------------|------------|
| 1      | English_Mail_Ratio (log) | Model 1 | 1.937 | 0.128 | 0.407 | 0.449 | 0.183 | -0.533 |
|        |                     |                   | (1.651) | (0.693) | (0.103)** | (0.465) | (0.129) | (0.479) |
| 2      | English_Mail_Ratio (log) | Model 2 | 1.677 | -0.961 | 0.236 | 0.408 | 0.073 | -0.274 |
|        |                     |                   | (2.221) | (0.803) | (0.130)** | (0.617) | (0.170) | (0.665) |
| 3      | New V Model |                   | 0.319 | 1.983 | 0.289 | 0.066 | 0.176 | -0.415 |
| 4      | * English_Mail_Ratio (log) | | (2.892) | (1.273) | (0.181) | (0.772) | (0.227) | (0.816) |

Note: N=91, ***p<0.01, **p<0.05, *p<0.10, White’s robust standard errors are used.
Finally, we find some evidence that the quality of communications between customer and team members affects the number of market defect density. But, this result should be interpreted with caution. If communicating with customers in English really causes some information to be lost in communication, why doesn’t it affect defect density in integration tests or unit tests, or even productivity. Earlier, we argued that a larger share of English emails enabled better information sharing with AMRD-US, which might have more than offset the negative effect.

This paper not only contributes to the literature of software engineering, but also to that of strategic management of R&D process and collaboration. First, the paper is the first case study that describes how the integral architecture concept is operationalized in the context of software development in the Japanese automobile industry. Second, it is the first paper that measures the impact on R&D productivity and product quality of a change in software development process from traditional sequential approach to new integrated approach that is becoming more common in the automobile ECU development.

One limitation of this study is the lack of direct measurement of functional performance. The paper asserts that CLIV has improved the functional performance at the car unit level by offering many opportunities for coordination with customers and other suppliers. The claimed improvement in functional performance, however, cannot be measured because the project teams develop products with different specifications and their functional performance cannot be compared directly. Although we used a sharp increase in specification changes as an evidence of improved functional performance, the number of specification changes only partially reflects the actual improvement and thus is a very noisy signal of functional performance. We believe it is necessary to invent a more comparable metric of functional performance to evaluate more accurate impact on product quality.

Another possible extension of the study is to consider what other practices or techniques in software development are complementary with CLIV. One such candidate is test-driven development (TDD). When TDD is introduced to the V model, a test case created for coding can be reused in the software test, which could reduce the workload for software tests. Furthermore, writing software codes following the test case substantially reduce the errors detected by the unit (software) test. These reductions in workload for software tests and potential errors detected by those tests are believed to make it easier to advance integration tests ahead of unit (software) tests at the time of switching from TIV to CLIV. The above discussion led us to conclude that there is a strong complementarity between TDD and CLIV. In future study, we plan to explore for proposing CLIV-TDD model and evaluate the impact of such new model.

References

[1] Abernathy, J., Clark, B. and Kantrow, M. (1983), “Industrial Renaissance: Producing a Competitive Future for America”, Basic Books
[2] Aoki, M. (1988), “Information, Incentives and Bargaining in the Japanese Economy: A Microtheory of the Japanese Economy”, Cambridge University Press
[3] Asanuma, B. (1990), “Manufacturer-Supplier Relationships in Japan and the Concept of Relation-Specific Skill”, Kyotu University - Keizai-ronsô, Vol. 145, No. 1-2, pp. 1-45
[4] Helper, S. (1991a), “Strategy and Irreversibility in Supplier Relations: The Case of the US Automobile Industry”, Business History Review, Vol. 65, No. 4, pp. 781-824
[5] Helper, S. (1991b), “How Much Has Really changed between U.S. Automakers and Their Suppliers？”, Sloan Management Review, Vol.15, pp.15-28
[6] Sako, M. (1991), “The role of ’Trust’ in Japanese buyer-supplier relationships”, Ricerche economiche, Vol. 45, No. 2-3, pp. 449-474
[7] Sako, M. (1992), “Price, quality and trust, Inter -firm relations in Britain and Japan”, Cambridge University Press
[8] Sako, M. and Helper, S. (1998), “Determinants of trust in supplier relations: Evidence from the automotive industry in Japan and the United States”, Journal of Economic Behavior and Organization, Vol. 34, No. 3, pp. 387-417
[9] Fujimoto, T., Takeishi, A. and Aoshima, Y. (2001), “Business Architecture: Strategic Design of Product, Organizations, and Processes”, Yuhikaku Corporation
[10] Nikkei Business Publications, Inc. (2005), “All about Car Electronics”, Nikkei Automotive Technology
[11] Iizawa, S. (2000), “Quality Management”, Unisys technology review, Vol. 67, pp. 69-87
[12] Mori, T., Sakuraba, N. and Nakano, T. (2006), “Development and Deployment of Software Quality Assurance Techniques”, Toshiba review, Vol. 61, No. 1, pp. 26-31
[13] Osuka, T. and Ogin, F. (2007), “Software Testing Technology”, Unisys technology review, Vol. 93, pp. 70-88
[14] Eguchi, K. (2009), “Software Engineering Supporting Advanced Software Development”, Toshiba review, Vol. 64, No. 4, pp. 32-35
[15] Boehm, B. (1981), “Software Engineering Economics”, Prentice-Hall
[16] Basili, V. and Perricone, B. (1984), “Software Errors and Complexity: An Empirical Investigation”, Communications of the ACM, Vol. 27, No. 1, pp 42-52
[17] Jeffery, D. and Lawrence, M. (1985), “Managing Programming Productivity”, Journal of Systems and Software, Vol.5, No.1, pp 49-58
[18] Briand, L., Differding, C. and Rombach, D. (1996), “Practical Guidelines for Measurement-Based Process Improvement”, Journal of Software Maintenance and Evolution: Research and Practice, Vol.2, No.4, pp 253-280
[19] Mizukami, Y. (2008), “A Challenge to an American Automotive Supplier to meet the Japanese Development Process for In-vehicle Embedded Software Development”, Information Processing Society of Japan, Symposium Proceedings, Vol. 2008, No. 9, pp 199-206
[20] Kline, S. J. (1985), “Innovation is not a Linear Process”, Research Management, Vol. 28, No. 4, pp. 36-46
Applying Integrated R&D Process in Process Innovation Research: Estimating the Impact of a Process Change in Automotive ECU Development on Organizational Flexibility and Product Quality

[21] Henderson, M. and Clark, B. (1990), “Architectural innovation: The reconfiguration of existing product technologies and the failure of established firms”, Administrative Science Quarterly, Vol. 35, pp. 9-30

[22] Nishiguchi, T. (1994), “Strategic Industrial Sourcing”, Oxford University Press

[23] Nobeoka, K. (1997), “Alternative Component Sourcing Strategies within the Manufacturer-Supplier Network: Benefits of Quasi-Market Strategy in the Japanese Automobile Industry”, Kobe Economic & Business Review, Vol.41, pp. 69-99

[24] Boehm, B. (1988), “A Spiral Model of Software Development and Enhancement”, Computer, IEEE, Vol. 21, No. 5, pp. 61-72

[25] Baldwin, C. and Clark, K. (2000), “Design Rules: The Power of Modularity”, MIT Press

[26] Ulrich, K. and Eppinger, S. (2000), “Product Design and Development”, McGraw-Hill

[27] Dustin, E., Rashka, J. and Paul, J. (1999), “Automated Software Testing: Introduction, Management, and Performance”, Addison–Wesley Professional

[28] Tobin, J. (1958), “Estimation of Relationships for Limited Dependent Variables”, Econometrica (The Econometric Society), Vol. 26, No. 1, pp. 24-36

[29] Beck, K. (2003), "Test-Driven Development", Addison-Wesley
