Title	Innovation Trajectories of Embedded Software : the Case of Automobiles and Mobile Phones in Japan
Author(s)	XIE, Zhongquan; MIYAZAKI, Kumiko
Citation	年次学術大会講演要旨集, 27: 321-325
Issue Date	2012-10-27
Type	Conference Paper
Text version	publisher
URL	http://hdl.handle.net/10119/11031
Rights	本著作物は研究・技術計画学会の許可のもとに掲載するものです。This material is posted here with permission of the Japan Society for Science Policy and Research Management.
Description	一般講演要旨

2B31

Innovation Trajectories of Embedded Software: the Case of Automobiles and Mobile Phones in Japan

○Zhongquan XIE[1,2], Kumiko MIYAZAKI[1]
(1. Graduate School of Innovation Management, Tokyo Institute of Technology; 2.The Institute for Integrated Cell-Material Sciences (WPI-iCeMS), Kyoto University.)

Abstract: This paper tries to contribute to the study on innovation trajectories of embedded software in the cases of embedded software in automobiles and mobile phones in Japan. By analysing the functions realized by embedded software divided into categories with different degrees of real-time computing and reliability, employing the patent data from USPTO and JPO, the paper found that innovation trajectories of embedded software are heterogeneous in different types of functions and there have been shifts of innovation trajectories in embedded software from control-centered to information-centered functions. Different types of interaction between computing technological paradigm and the paradigm of related technologies in functions with different degrees of real-time computing and reliability are basically technological factors influencing innovation trajectories of embedded software. Implications related to innovation trajectories and innovation strategies of embedded software are given.
Keywords: Embedded software, innovation trajectory, automotive software, mobile phone, Japan

1. Introduction

Innovation is advanced in certain directions and pathways along the entire value chain from R&D to commercialization. Such directions and pathways are called innovation trajectories. Shifts in innovation trajectory has been identified as significant factor resulting in the structural changes in the national systems by Kumaresan and Miyazaki [1] with an empirical analysis of the Japanese robotic industry. Analysis of innovation trajectory in different sectors has been identified as a significant field in innovation studies (e.g., in ref. [2-5]).

Embedded software is a kind of software that embedded in specific machines/devices/equipment/ apparatus (e.g., robots, automobiles, digital appliances and mobile phones) to realize specific functions. Generally, embedded software is implemented (e.g., built in Read Only Memory) by the manufacturers during the manufacturing process and it cannot be added, modified or deleted by users in principle. As a "source of international competitiveness" of Japan [6], embedded software plays a significant role in economic development and social welfare.

Based on both the theoretical and empirical significance, the research on innovation trajectories of embedded software can provide significant meanings for innovation management and strategies for embedded software. However, until now, there has been a lack of research on innovation of embedded software, besides the studies on US and European embedded software sectors in terms of research activities and software tools [7, 8], research activities in Asian embedded software [9], open innovation processes in the case of embedded Linux [10], product innovation of embedded software in the case of automotive software [11] and mobile phones [12].

This paper tries to address the following two research questions: (1) How is the innovation trajectory of embedded software? (2) What technological factors result in such innovation trajectory?

While Kumaresan and Miyazaki (2001) argued that innovation trajectories should be analyzed integratedly rather than independently in terms of markets or technologies [1], in this short paper, we only presents the technological related product innovation trajectory of embedded software, especially in terms of the functions realized by embedded software. The function here is a broad concept defined as purpose of embedded software, such as being used to "control, monitor, or assist" [13] specific machines/devices/equipment/apparatus in a digital way, neither a set sequence of steps, part of larger computer program nor having a meaning in mathematics.

The paper is organized as follows: Section 2 discusses the theoretical background, analytical framework, methodology and data. Section 3 presents the main findings. Section 4 provides conclusions and implications.

2. Theoretical Background, Analytical Framework, Methodology and Data

As a semi-finished product, the functions of embedded software are realized by embedded into specific embedded products (e.g., automobiles or mobile phones). In such sense, product innovation of embedded software is defined as *the first attempt to embed software into an embedded product, or a significant improvement in the software for an embedded product*. The functions realized by embedded software are considered as a proxy of product innovation of embedded software. Therefore, the product innovation trajectory of embedded software is understood by investigating the functions realized by embedded software.

The functions realized by embedded software put specific requirements on embedded software, since embedded software is a "design for real world" [8], "not the transformation of data, but rather the interaction with the physical world", and "of necessity, acquire some properties of the physical world" [14]. The "interaction with the physical world" also indicates that there are different types of interactions between *technological paradigms* (defined as "the 'model' and a 'pattern' of solution of selected technological problems" by Dosi [15]). Therefore, such interactions between technological paradigms indicate heterogeneous innovation trajectories, because of different requirements in types of interactions to realized specific functions. In other words, different requirements on embedded software due to the interactions with physical world are basically technological reasons that heterogeneous innovation trajectories of embedded software exist.

Such different requirements on embedded software involve two significant aspects: *real-time computing* and *reliability*. *Real-time* programs must guarantee a response within strict time constraints [16], such as in milliseconds or microseconds. The issue is not just that execution takes time; even with infinitely fast computers, embedded software would still have to deal with time because the physical processes, with which it interacts, evolve over time [14]. Deadlines of real-time tasks can be classified as either *hard, firm,* or *soft* [17]. A deadline is said to be *hard* if missing a deadline is a total system failure or catastrophic; e.g., if software in anti-lock braking system (ABS) cannot meet the deadline response time, automobile may be likely to enter a spin which could result in an accident. *Firm real-time* means that infrequent deadline misses are tolerable, but may degrade the system's quality of service and consequences of not meeting the deadline are not very severe, e.g., software realizing call function in mobile phones. *Soft real-time* means that the utility of results produced by a task with a soft deadline decreases over time after the deadline expires, e.g., software that maintains and updates the flight plans for commercial. The different degrees of real-time computing requirements on embedded software demand different interactions between computing technological paradigm and the paradigms and the technological paradigms of related technologies. These different requirements on real-time computing may result in heterogeneous innovation trajectories.

Reliability means that embedded software systems are expected to run continuously for years without errors. In some cases the systems recover by themselves if an error occurs, e.g. embedded software system in space systems, undersea cables, navigational beacons, bore-hole systems, and some sub-systems of automobiles. In some cases, accident may happen if an error occurs, such as in the embedded software for controlling car engine, aircraft navigation, reactor control systems and train signals. This means that embedded software is usually developed and tested more carefully than that for personal computers and a variety of techniques are needed to deal with the errors or reduce the errors. The requirement of reliability increases the burden of embedded software engineers, also results in heterogeneous innovation trajectories.

Therefore, we propose:

Hypothesis 1 - *Innovation trajectories of embedded software are heterogeneous in different types of functions.*

Hypothesis 2 – *The different requirements on embedded software due to interactions between various technological paradigms, such as real-time computing and reliability are basically technological reasons of heterogeneous innovation trajectories of embedded software.*

To test the hypotheses, we select the cases of embedded software in automobiles and mobile phones in Japan, since embedded software realized functions in these two domains cover different degrees of real-time computing and reliability. The functions realized by automotive software are divided into the following five categories: (i) driving force control (power train control); (ii) driving force control (battery and electric power control); (iii) safety control; (iv) body control; and (v) information communication and telecommunication (IC&T) system [11]. The functions of embedded software in mobile phones are divided into the following categories: (i) call related communication functions; (ii) non-call related communication functions; (iii) functions related to hardware component control; and (v) data related functions [12]. Especially, the functions realized by the application software embedded in mobile phones to connect internet or to provide services through internet, such as based on cloud computing, are named as "internet connection" and classified into "data related functions". The different fields of functions in dimensions of real-time computing and reliability are presented roughly in Fig. 1.

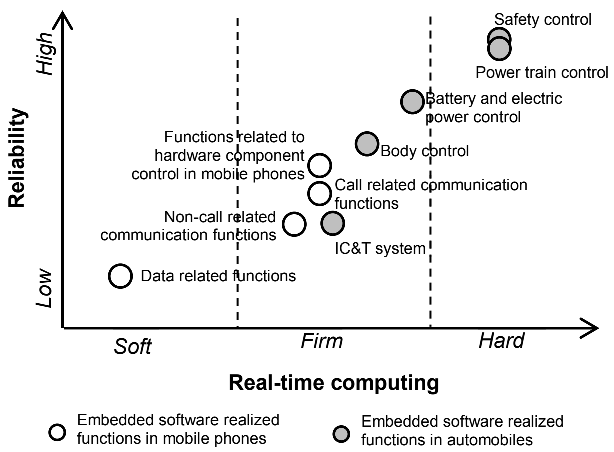

Fig. 1 Proposed characteristics matrix of different fields of functions

Subsequently, we analysis the innovation activities in different fields of functions to investigate whether their innovation trajectories are different and to identify what kinds of technological factors are responsible for such heterogeneity.

Innovation activities in embedded software realized functions in automobiles are measured by the patents issued in United States Patent and Trademark Office (USPTO). The functions realized by embedded software in mobile phones are measured by patents that claimed as a mobile phone in Japan Patent Office (JPO). The results are presented in the next section.

3. Main Findings

3.1. Heterogeneous innovation trajectories of embedded software

Fig. 2 presents the innovation activities of Japanese actors in automotive software realized functions indicated by issued USPTO patents assigned to Japanese applicants. Safety control has become the top focus of automotive software innovations since the emergence of car electronics and evolved in the same way as total automotive software related patents. However, automotive software related patents related to IC&T systems kept increasing compared to others, and surpassed power train and body control in the mid-1990s, becoming the second important area of product innovation of automotive software. Battery and electric power control has been going up steadily during the last three decades. The increasing use of electronic control units (ECUs) and demand for hybrid and electric cars highlighted the significance of battery and electric power control. Comparatively, the numbers of yearly issued automotive software related patents on power train and body control have become relatively stable after the mid-1990s.

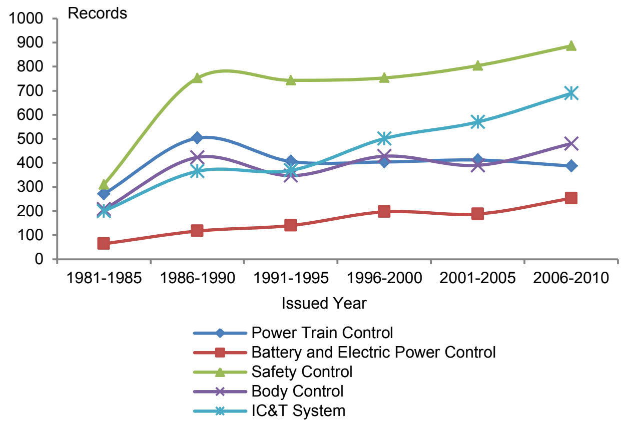

Fig. 2 Different innovation activities in automotive software realized functions indicated by issued USPTO patents assigned to Japanese applicants

Source: Xie and Miyazaki (2011)[11]

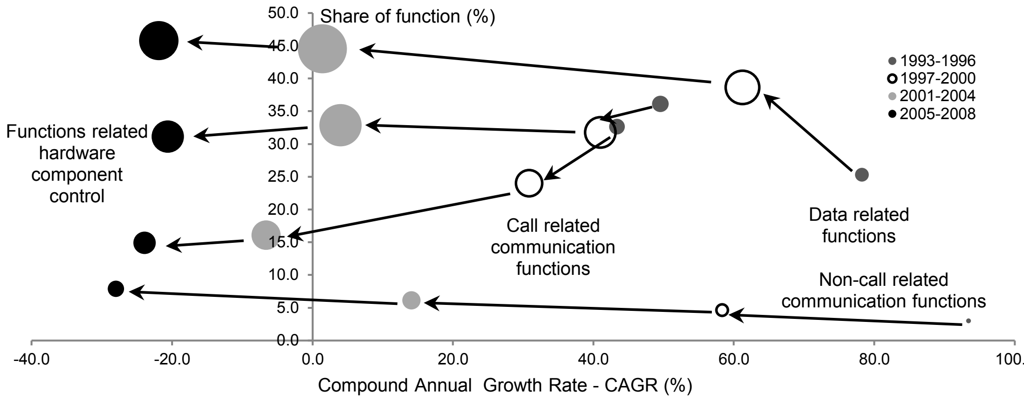

Fig. 3 Evolutionary patterns of innovation activities by types of functions realized by embedded software in mobile phones indicated by patent applications by Japanese applicants in JPO

Source: Xie and Miyazaki (2012)[12]

Fig.3 depicts the changes of the share of functions and compound annual growth rate (CAGR) of innovation activities in specific functions in mobile phones during different periods. It shows that there have been increases of share of functions in the field of non-call related communication functions and data related functions; and decreases of share of functions in the field of call related functions and the functions related to hardware component control. This indicates the increasing significance of data related functions in mobile phones. Fig.3 also shows that innovative activities in embedded software in mobile phones followed a certain trajectory: increase at an early stage (before 2000, especially 1998-2000), relative stability in the middle stage (2000-2004), and a decrease at its final stage (after 2004).

Comparing the two cases, we obtained the following findings:

(1) The share of innovation activities on the functions related to information processing, such as IC&T system and data related functions have increased during the last ten years. This indicates that data related functions and IC&T system have been playing more and more significant role as innovation focus in embedded software realized functions.

(2) The innovation activities of embedded software in the field of functions related to control specific devices/apparatus in embedded products did not increased, in corresponding to the maturity of these technologies, such as the table innovation activities in the field of power train control in automobiles and the decrease of call related functions and the functions related to hardware component control in mobile phones.

(3) The software functions related to controlling new emerging technologies increased, which are corresponding to the increasing need of functions realized by embedded software, such as battery and electric power control in automobiles and non-call related communication functions (near field communication-NFC, 1sg television, etc.). This indicates that the interaction of computing technological paradigms and the paradigms

related to specific technologies are growth points of innovation activities in embedded software.

(4) There have been shifts of innovation trajectories in embedded software. If we consider power train control in automobiles, call and non-call related functions and the functions related to hardware component control in mobile phones as control-centered functions; and IC&T system and data related functions as information-centered functions, then we can say that there has been a shift of innovation trajectories in embedded software from control-centered to information-centered functions. Furthermore, if the control-centered functions are still at its early stage, such as the functions related control near field communication (NFC), there should be some concentration on such control-centered functions. Once such control-centered functions has been realized to some extent, there is requirement on focusing more on information-centered functions.

In summary, there are different innovation focuses and compound annual growth rates in product innovation of embedded software. This supports **Hypothesis 1**. In Section 3.2, we will continue to discuss the technological reasons of such heterogeneities of product innovation trajectories in embedded software.

3.2. Technological reasons of heterogeneous innovation trajectories

As seen in Fig. 1, functions are presented in terms of real-time computing and reliability. The characteristics of these functions are considered as reasons technological possibilities of reasons related to heterogeneous innovation trajectories. The development of related technologies has met the requirements of embedded software related to different functions, resulting in different innovation trajectories. We presume that the realization of embedded software realized functions (namely, innovation of embedded software) depends on the maturity of related technologies and computing technologies to some extent. We analyze the reasons as follows:

(1) The functions of safety control and power train control in automobiles are characterized by hard real-time computing and high reliability. Hard real-time computing means that there is a pressing need for engineers to consider the close relationship between computing technological paradigm and the technological paradigm of power train control, which implies a specific innovation pattern of embedded software in such types of functions. Meanwhile, high reliability requires much more consideration on embedded software to make it as few bugs as possible. Considering these two characteristics of such type of embedded software realized functions, we could understand that there is no significant change in innovation activities in these two functions since the 1990s, while the processors' speed has increased dramatically. This indicates that innovation trajectory of embedded software realized functions with the characteristics of hard real-time computing and high reliability depends more on the technological paradigm of related technologies rather than computing technological paradigms.

(2) The functions of battery and electric power control, body control and IC&T systems in automobiles, call and non-call related communication functions, and functions related to hardware component control in mobile phones are characterized by firm real-time computing and middle level of reliability. This kind of functions depends equally to some extend on computing technological paradigm and the paradigm of related technologies, such as telecommunication and control. Then, there are three cases: (i) Related technologies are more developed than computing technology at first; and then the development of computing technology meets the requirements of related technologies. The embedded software realized functions of body control in automobiles, call related communication in mobile phones are the representative of this case. (ii) Both computing technology and related technologies are still under development at first; and then the development of both computing technology and related technologies to some extent make realization of specific function be possible. This case refers to the functions battery and electric power control, IC&T system in automobiles, non-call related communication, functions related to hardware component control in mobile phones. (iii) Computing technology is almost mature, but related technologies are still under development. Once the related technologies become much more mature, innovation in such field would become significant. The function of battery and electric power control in automobiles is located in this case. The increasing innovation activities in this function indicate that innovation in this field is at its early stage. In summary, in functions equally depending on computing technological paradigm and the paradigm of related technologies, the maturity of these paradigms to some extent is the determinative factor of innovation trajectory of embedded software in such field.

(3) The data related functions are characterized by soft real-time computing and low reliability. This kind of functions is highly depending on computing technological paradigm rather than the paradigm of related technologies. In other words, the development of computing technology plays a determinative role. In the case of mobile phones, innovation in data related functions are determined by processors' speed to large extent. Xie and Miyazaki (2012) have already argued that the increasing speed of processors is a main factor resulting in the increase of data related functions [12]. This indicates that innovations in the functions with soft real-time computing and low reliability are more likely depending on the development of computing technology.

In summary, different types of interaction between computing technological paradigm and the paradigm of related technologies in functions with different degrees of real-time computing and reliability are basically technological factors influencing innovation trajectory of embedded software, which supports **Hypothesis 2**.

4. Conclusions and Implications

This paper discussed the innovations trajectories of embedded software by employing the cases of embedded software in automobiles and mobile phones. The embedded software realized functions were divided into categories with different degree of real-time computing and reliability. The innovation activities indicated by the functions realized by embedded software were analyzed according to the categories, which indicate heterogeneous innovation trajectories. The paper analyzed that different types of interaction between computing technological paradigm and the paradigm of related technologies in functions with different degrees of real-time computing and reliability are basically technological factors influencing innovation trajectory of embedded software.

There are implications based on these results. Firstly, different innovation trajectories in embedded software in terms of functions imply different innovation strategies in embedded software. Implementing innovation strategies should consider the following aspects: (a) Significance of specific functions. (b) Technological characteristics related to specific field of functions with different requirements of high accumulation of specific knowledge or real-time computing. (c) Different stages of innovation trajectories.

Secondly, the innovation trajectory shift from control-centered to information-centered implies a significant change of the role of embedded software. Therefore, there is pressing need for engineers to change the viewpoints from hardware-dominated to software-dominated development process and to innovate in the development process to meet the needs related to the shift of innovation trajectory.

Thirdly, the result that product innovations of embedded software in different types of functions were the results of interaction among technological paradigms along innovation trajectories has two implications: (a) New connection between software engineering technologies and other devices may provide new innovation trajectories in terms of functions realized by embedded software. (b) The co-evolution of technological paradigms also implies cooperation between software engineers and the engineers related to specific technologies. Product innovations occur when engineers cooperate to make embedded software realize specific functions.

Fourthly, understanding the different innovation trajectories of embedded software depending on how heavily depend on computing technological paradigm and the paradigm of related technologies may help engineers or companies making appropriate innovation strategies for embedded software, according to the degree of real-time computing and reliability of the embedded software realized functions.

This paper mainly discussed technological reasons related to heterogeneity of innovation trajectories, not mentioning non-technological reasons or the activities of innovative actors, since the page limitation. Such kind of research topics will be discussed in our future study.

Reference

[1] N. Kumaresan, and K. Miyazaki, “Management and Policy Concerns over shifts in Innovation Trajectories: The Case of the Japanese Robotics Industry,” *Technology Analysis and Strategic Management* vol. 13, no. 3, pp. 433-462, 2001.

[2] P. E. Hart, “Artificial Intelligence in Transition,” *AI Magazine,* vol. 5, no. 3, pp. 17-20, 1984.

[3] P. Quintas, “Programmed Innovation? Trajectories of Change in Software Development,” *Information Technology & People,* vol. 7, no. 1, pp. 25-47, 1994.

[4] F. Hacklin, V. Raurich, and C. Marxt, “Implications of Technological Convergence on Innovation Trajectories: The Case of ICT Industry,” *International Journal of Innovation and Technology Management (IJITM),* vol. 2, no. 3, pp. 313-330, 2005.

[5] M.-C. Hu, and C.-Y. Wu, “Exploring Technological Innovation Trajectories through Latecomers: Evidence from Taiwan's Bicycle Industry,” *Technology Analysis and Strategic Management,* vol. 23, no. 4, pp. 433-452, 2011.

[6] METI, "Government Initiatives for the Competiveness Strengthening of Japanese Software," 2004.

[7] W. E. Steinmueller, "The European Software sectoral system of innovation," *Sectoral Systems of Innovation: Concepts, Issues and Analysis of Six Major Sectors in Europe*, F. Malerba, ed., Cambridge: Cambridge University Press, 2004.

[8] W. E. Steinmueller, “Embedded Software: European Markets and Capabilities,” European Sectoral Systems of Innovation Project Working Paper, SPRU, University of Sussex, 2001.

[9] K. Miyazaki, and K. Klincewicz, "Sectoral Systems of Innovation in Asia: The Case of Software Research Activities." pp. 726-731.

[10] J. Henkel, “Selective revealing in open innovation processes: The case of embedded Linux,” *Research Policy,* vol. 35, no. 7, pp. 953-969, 2006.

[11] Z. Xie, and K. Miyazaki, "Dynamics and heterogeneity of product innovation in embedded software: The case of Japanese automotive software.." pp. 1-13.

[12] Z. Xie, and K. Miyazaki, “Product Innovations of Embedded Software in Mobile Phones: An Emprical Analysis Based on the Functions Realized by Embedded Software,” *Journal of Information and Communication Research,* vol. 30, no. 1, pp. 27-42, 2012.

[13] IEEE, “Embedded Systems.,” in Embedded Systems and the Year 2000 Problem: Guidance Notes., IEE Technical Guidelines 9, 1995, pp. 15-27.

[14] E. A. Lee, "Embedded Software," *Advances in Computers*, 56, M. Zelkowitz, ed., pp. 55-95, London: Academic Press, 2002.

[15] G. Dosi, “Technological paradigms and technological trajectories: A suggested interpretation of the determinants and directions of technical change,” *Research Policy,* vol. 11, no. 3, pp. 147-162, 1982.

[16] M. Ben-Ari, *Principles of Concurrent and Distributed Programming*, p.^pp. 164: Prentice Hall, 1990.

[17] K. G. Shin, “Real-Time Computing: A New Discipline of Computer Science and Engineering,” *Proceedings of the IEEE,* vol. 82, no. 1, pp. 6-24, 1994.