Eurasian Journal of Economics and Finance, 8(4), 2020, 274-278
DOI: 10.15604/ejef.2020.08.04.007

EURASIAN JOURNAL OF ECONOMICS AND FINANCE

www.eurasianpublications.com

PROBLEMS OF SMALL AND MID-SIZED ENTERPRISES IN JAPAN'S SOFTWARE INDUSTRY

Mita Takahashi
Osaka Sangyo University, Japan
Email: takahashi@dis.osaka-sandai.ac.jp

Received: November 4, 2020 Accepted: December 21, 2020

Abstract

This paper examines the reasons that most small and mid-sized enterprises (SMEs) in Japan's software industry do not develop into large firms. Thousands of SMEs are informally organized under several large companies. Those SMEs have experienced difficulty developing into large firms. This industrial structure has remained for several decades. Several papers have already suggested reasons for this phenomenon, but each paper in the literature has given few reasons. This paper analyzes the issue more comprehensively. First, this article surveys the literature and determine several factors that contribute to the issue. Then, this paper demonstrates how they complicate the issue and make it difficult to solve. Finally, the paper illustrates several approaches in an effort to find solutions. The first method is introducing agile modeling and other iterative software development processes. Agile modeling adjusts design change more easily compared with other software development processes. Other iterative software development processes also adjust design change easily compared to the waterfall model. By using these processes, software engineers can avoid long overtime work, which contributes to the difficulties in securing human resource. The second method is for software firms to demand appropriate compensation for each design change. If many Japanese firms did this, overtime work would be shorter, and the working environment of SMEs would improve. Then, SMEs obtain excellent workers. This will help SMEs to grow to large firms.

Keywords: Software Industry, SMEs, Japan, Custom Software

1. Introduction

The Japanese software industry works within a large subcontracting system. Thousands of small and mid-sized enterprises (SMEs) are informally organized under several large companies. Those SMEs have experienced difficulty developing into large firms. In addition, as Baba *et al.* (1996) said, the software SMEs were not able to develop their own products. Porter *et al.* (2000) also stated Japanese customers resisted paying for pre-packaged software. This industrial structure has remained for several decades.

Several authors have taken up this issue in the past; however, each paper in the literature has given only one (or two) reason for the failure of SMEs to expand to become large companies. These papers failed to analyze the issue comprehensively. Therefore, the purpose of this paper is to reveal the reasons that most small and mid-sized enterprises (SMEs) in Japan's software industry do not develop into large firms.

In general, Japanese SMEs have difficulty in developing into large firms compared to other countries. In Japan, many firms prefer the long-time and repeated transactions with the limited number of firms to the transaction with the variety of firms. Small firms, therefore, cannot easily expand their transaction partners. This characteristic is very unique compared to foreign countries (Bilgin *et al.* 2012), and true for software SMEs in Japan. However, Japanese software SMEs have to face larger obstacles as this paper explains.

This paper uses the following methodology. First, this paper surveys the literature, and proposes five reasons why software SMEs have experienced difficulty in developing into large firms. Then, I show the causal connections among these five factors, demonstrating how these connections complicate the issue and make a solution difficult to identify. Through this analysis, I examine the issue comprehensively and prove that factors that contribute the issue have causal connections with each other. These connections complicate the issue and make it difficult to solve, and these factors as a whole prevent software SMEs from developing into large firms. The preceding literature has not fully recognized this point.

This causal connection and related findings are the main contribution of this paper. Through the analysis, this paper describes why many software SMEs in Japan cannot grow for a long time.

This paper focuses on software firms, which develop packaged or custom software. I do not aim to analyze embedded software or game software, as embedded software is simultaneously developed with the hardware of manufactured goods. The game software industry is largely influenced by the Japanese animation culture. Therefore, I distinguish them from packaged and custom software. However, some factors described in this paper hold true to embedded and game software in Japan.

This paper is organized as follows. In Section 2, I survey the literature and determine five factors that contribute to the issue. Section 3 shows the causal connections among these five factors, demonstrating how these connections complicate the issue and make it difficult to solve. Section 4 concludes this paper by illustrating several approaches in an effort to find solutions.

2. Five factors preventing software SMEs from developing

2.1. The division of the market

The first factor, the division of the market, stems from the business environment in Japan's computer industry. Until the early 1990s, Japanese computer manufacturers had been developing both hardware and software. Since the software worked only on the hardware developed by the same manufacturer, it had to be specially developed for the hardware of each manufacturer. This resulted in the partitioning of the software market into many small markets. For the software SMEs, segmentation makes it difficult to spread high fixed costs, so they were not able to develop their own products (Baba *et al*. 1996).

Since the middle of 1990s, software has worked for hardware made by any computer manufacturer. Therefore, the segmentation does not exist anymore. However, the next section shows that this factor still influences the Japanese software market.

2.2. The inappropriate valuation of software

The second factor is that the value of software is not appropriately evaluated, and software development is therefore inappropriately priced. This situation is described by Porter *et al.* (2000) who said Japanese customers strongly preferred custom software, which was often bundled with mainframe computers by Japanese computer companies. Because they expected software to come 'free' with the computer, Japanese customers resisted paying for pre-packaged software separately. That inhibited development of the much more cost-effective software solutions that were widely accepted in the United States and elsewhere.

2.3. Problems related to securing human resources

The third factor involves the problems that software SMEs encounter when seeking reliable and

efficient human resources. A major reason for this is terrible working environment of the Japanese software industry. The Japanese software industry is often described as hard work, going home late at night or even sometimes sleeping at the workplace, and low wages. A survey shows that the largest problem for IT (Information Technology) firms in recruiting newly graduating students is “the bad image of the industry” (Information-Technology Promotion Agency, Japan, 2008). In addition, highly-skilled engineers are not always paid nor promoted appropriately. This custom hampers software engineers’ motivation to acquire more skills. These conditions negatively affect the ability of the software industry to recruit top students, and cause engineers working for software firms to transfer to other industries.

2.4. Problems related to universities

The fourth factor is problems related to Japanese universities. As the content of research of software engineering at Japanese universities is far from considered practical use, research in universities rarely contributes to software SMEs, which lack ample funds for their own research and development.

Japanese university education also has several problems. In the 1990s, there existed a lack of professors of computer science to a large extent. Since then, there has been an increase in these positions; however, their education is not towards any practical use compared with the United States, India, and other countries (Yamashita, 2007). As a result, most Japanese newly-hired software engineers cannot carry out the work of software firms.

2.5. The lack of able engineers

The fifth factor is the lack of ability and motivation among Japanese software engineers. Matsubara (2001), a former engineer at Hitachi, Ltd. attributed to a lack of “creative ideas and products” among the software community. In addition, the Software Global Competitiveness Research Committee, established by the Institute of Electronics, Information and Communication Engineers in Japan, has stated that the motivation of Japanese software engineers to capture market is weaker than foreign engineers (Matsumoto, 1999).

3. Causal connections of factors preventing software SMEs from developing

In this section, I show the causal connections among these five factors, and demonstrate how these connections complicate the issue and make its solution evasive.

3.1. The reason why the inappropriate valuation of software is caused

I now consider why the second factor, the inappropriate evaluation of software is caused. As I described in the former section, Japanese computer software had to be specially developed for the hardware of each manufacturer. This business environment resulted in the first factor, the partitioning of the software market into many small markets.

Because of the partitioned market, there was not much competition among computer manufacturers. They raised prices of hardware and earned excess profits from hardware sales, so they tended to handle software as an add-on of sorts, selling it for very low prices. Consequently, Japanese software clients did not place an accurate value on the software.

The segmentation of the market disappeared in the middle of 1990s. However, the Japanese clients still inappropriately evaluate the value of software. In other words, they do not understand the value of software even today. So, they try to pay less for software. One of the reasons why this tendency continues is that Japanese customers consider software as just a cost. They do not think software as a source of competitiveness. So, they try to pay less for software. This situation indicates that the first factor causes the second factor.

3.2. The cause of the problem related to securing human resources

I now consider the cause of the third factor, problems related to securing human resources. Because software clients have failed to understand the value of software, large software firms in Japan have needed to decrease the cost of developing software by using software SMEs as subcontractors. This practice has resulted in the establishment of a huge subcontracting system in the Japanese software industry.

This subcontracting system causes a further problem. The system is suitable for the waterfall model, which is the most widely-used software development process in Japan. The model supposes that each phase of the software development be perfectly determined and never changed afterward. However, because Japanese clients cannot decide what kind of software they need, they often revise the design of software after it is almost completely developed. This kind of large-scale design change causes significant overtime work for software engineers. This then leads to problems related to securing human resources. This situation indicates that the second factor causes the third factor.

3.3. The cause of the lack of able engineers

Next, I consider the causes of the fifth factor, the lack of able engineers. As I discussed under the third factor, significant overtime hours is common in Japan's software industry. These working conditions and low wages lead to the decline of the industry's image. In addition, the fourth factor shows that the research and education of universities is far from involved in practical use. These things contribute to the lack of able engineers. This situation indicates that the third and fourth factors cause the fifth factor.

4. Conclusion

This paper examined the reasons why most software SMEs in Japan's software industry have failed to develop into large firms. I determine five factors that contribute to the issue—(1) the division of the market, (2) the inappropriate valuation of software, (3) problems related to securing human resources, (4) problems related to universities, and (5) the lack of able engineers.

This paper also shows the causal connections among these factors. The first factor causes the second factor, which causes the third factor. In addition, the third and fourth factors cause the fifth factor. The first factor, the partitioning of the software market into many small markets, resulted in excess profits for hardware sales and lower prices for software. This custom caused the second factor, the inappropriate valuation of software by clients.

Because of the inappropriate valuation of software, large software firms in Japan have needed to decrease the cost of developing software. They exploit the many software SMEs as subcontractors and use the traditional software development process. However, the development system does not suit the behavior of Japanese clients, resulting in prolonged overtime work for software engineers. This situation results in problems related to securing human resources.

Because of problems related to securing human resources and universities, the software industry has encountered difficulties in recruiting motivated engineers and encouraging them to obtain practical knowledge and skills. These circumstances contribute to the lack of able engineers. These connections complicate the issue and make it difficult to solve. This is why most SMEs in Japan's software industry do not develop into large firms. As a result, this industrial structure has remained for several decades.

This paper's policy implications are that the Japan's software industry needs some new approaches in their development methods or transactions. In our history, new technology or transaction will affect the present situation of the market. So new approaches will change the economic situation and solve some problems in Japan's software industry.

This paper illustrates several approaches that could be applied to new software development processes in order to solve this issue. As an example, I now take up the problem of large-scale design change by Japanese clients, which leads to the third factor. I suggest two methods of dealing with this problem.

The first method is introducing agile modeling and other iterative software development processes. Agile modeling adjusts design change more easily compared with other software development processes. Other iterative software development processes also adjust design change easily compared to the waterfall model. By using these processes, software engineers can avoid long overtime work, which contributes to the difficulties in securing human resources[1].

It is sometimes said that the agile modeling is not suited to large-scale projects. In fact, there are so many projects that hundreds of engineers participate in using the agile model.[2] Therefore, the agile model and other iterative software development processes are applicable to various projects.

The second method is for software firms to demand appropriate compensation for each design change. When Japanese clients change designs, they usually pay smaller compensation compared to the amount of the task. Therefore, software firms must reduce their costs for changing the design by decreasing the number of software engineers for the task. This is one of the reasons for long overtime work. However, some foreign software firms in Japan already demand appropriate compensation from their clients for design changes. If many Japanese firms did this, overtime work would be shorter.

Successful cases through these methods would create new business models, which would set Japanese software firms free from the above causal connections. Firms and governments must cooperate with each other to produce these trends.

References

Baba, Y., Takai, S., and Mizuta, Y., 1996. The user-driven evolution of the Japanese software industry: The case of customized software for machines. In: Mowery, D. C., ed. 1996. *The international computer software industry*. New York: Oxford University Press. pp. 104-130.

Bilgin, M. H., Lau, H. K. M., and Demir, E., 2012. Technology transfer, finance channels, and SME performance: New evidence from developing countries. *Singapore Economic Review*, 57(03), pp. 1-20. https://doi.org/10.1142/S0217590812500208

Information-Technology Promotion Agency, Japan, 2008. *The 29th Information-Technology Industry Management Report: Survey Implemented in 2007*. Tokyo: Information-Technology Promotion Agency, Japan.

Matsubara, T., 2001. Japan: A huge IT consumption market. *IEEE Software*, September/October, pp. 77-80. https://doi.org/10.1109/52.951499

Matsumoto, M., 1999. Does Japan work on the forefront in the century of information and communication? The Institute of Electronics, Information and Communication Engineers in Japan National Convention Articles, March.

Porter, M. E., Takeuchi, H., and Sakakibara, M., 2000. *Can Japan compete?* Cambridge, MA: Perseus Publishing.

Takahashi, M. and Takahashi, N., 2013. Unintended technology transfer to Chinese software firms from Japan through offshore software development. *Information Technologies & International Development*, 9(4), pp. 1-18.

Takahashi, N., 2017. Japanese ICT firms' offshore development and software business in East Asia. In: H. Hirakawa, N. Takahashi, F. C. Maquito, and N. Tokumaru eds., 2017. *Innovative ICT industrial architecture in East Asia: Offshoring of Japanese firms and challenges faced by East Asian economies*. Tokyo: Springer. pp. 61-78. https://doi.org/10.1007/978-4-431-55630-5_3

Yamashita, T., 2007. *Proposals of education and training of highly skilled IT human resources*. Tokyo: NHK Publishing.

[1] However, in Japan, the offshore development has been expanding (Takahashi and Takahashi, 2013; Takahashi, 2017). Agile development is not easily implemented with offshore development.

[2] IBM Japan started positively exploiting agile modeling in 2006. The firm implemented more than 320 projects using the model for three years until 2009.