

**This is an electronic reprint of the original article.  
This reprint *may differ* from the original in pagination and typographic detail.**

**Author(s):** Ojala, Arto

**Title:** Selection of the Proper Revenue and Pricing Model for SaaS

**Year:** 2014

**Version:**

**Please cite the original version:**

Ojala, A. (2014). Selection of the Proper Revenue and Pricing Model for SaaS. In J. Guerrero (Ed.), Proceedings of IEEE CloudCom 2014 : 6th IEEE International Conference on Cloud Computing Technology and Science (pp. 863-868). IEEE. doi:10.1109/CloudCom.2014.27

All material supplied via JYX is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

# Selection of the Proper Revenue and Pricing Model for SaaS

Arto Ojala

Department of Computer Science and Information Systems  
University of Jyväskylä  
Jyväskylä, Finland  
arto.k.ojala@jyu.fi

**Abstract** - Recent research on software revenue and pricing models has revealed important ways in which firms can benefit from software renting. However, it is still unclear how SaaS providers select a proper revenue and pricing model to make their services attractive for customers. Based on 32 interviews with software professionals from four case firms, this study reveals how different factors impacted on the selection of a revenue and pricing model. It can be concluded that customers' needs were the main driving force to the selection of the most appropriate pricing and revenue model in the market.

**Keywords** – Revenue models, software pricing, cloud computing, SaaS

## I. INTRODUCTION

The importance of cloud computing and related services have attracted increasing interest in information technology literature [1]. Cloud computing is changing the way in which software is delivered, sold, and used. It brings several opportunities for software vendors, but it also challenges existing software architectures and ways of doing business. To compete in the software market, vendors have to take these changes into the consideration and rethink how to offer their software to customers. When the software is sold through the software as a service (SaaS) model, it facilitates a number of revenue and pricing models. However, because there are many technical changes to consider, business-related issues are easily overlooked, making it difficult to sustain a profitable revenue stream.

The traditional way to sell the software license is to sell a perpetual software license for a single user or machine or to sell a license to use the software in a certain number of processors [2]. Recently, a number of studies have suggested that software renting is becoming more frequent in the new era of computing, in which software is delivered via the SaaS model [3, 4, 16].

Economic theories have been applied to the benefits of renting compared to purchasing a product outright [5, 6]. However, most of the existing literature [3, 7, 8, 9] on software revenue models has used analytical approaches that simplify real-world settings, seeking to apply algebraically testable rules to determine the benefits of software renting over the licensing model. Furthermore, even these studies have revealed important aspects of software renting vs. licensing, the focus of these studies have been on monopolistic market situations that neglect the free market competition [3, 7, 8, 9].

With these considerations in mind, the aim of this article is study how different factors in the market interact when a software firms make decision on the revenue and pricing models. In addition, the article builds on economic theories [5, 6] and previous work using analytical approaches to study revenue models used by software firms [3, 7, 8, 9]. The study reported in this article addresses the following main question: *How do different factors impact on SaaS providers' selection of revenue and pricing models?*

## II. LITERATURE REVIEW

### A. Cloud computing and SaaS

Cloud computing provides access to computing resources, storage space, and software applications via the Internet as a service. Cloud computing can be divided roughly into three service layers. These consist of (i) Infrastructure as a Service (IaaS), which provides computation and storage capacity, (ii) Platform as a Service (PaaS), which provides software development tools plus an application execution environment, and (iii) Software as a Service (SaaS), which provides applications on top of PaaS and IaaS [10].

The SaaS model evolved from Application Service Provisioning (ASP) in the late 1990s. ASP was developed as an alternative to on-premise software. It offered the possibility for clients to outsource the hosting and maintenance of the software to an ASP vendor [11]. The ASP model was based on a single-tenant architecture in which each customer had a customized version of the software in the ASP provider's server [12]. In contrast, SaaS has multi-tenant or multi-instance architecture; the former refers to the situation in which a vendor hosts a single instance on a server that serves multiple customers (tenants), whereas the latter is related to the model in which a vendor hosts separate instances for each customer within shared hardware [13].

SaaS providers can be classified into "pure-SaaS" and "enterprise-SaaS" business archetypes [14]. Pure SaaS refers to software that is simple to use and has low or no requirements for customization [15]. According to Benlian et al. [14], pure-SaaS products also have lower strategic significance in a customer's business processes compared to enterprise-SaaS products. In addition, Benlian et al. [14] found that pure-SaaS products, such as office systems, may have lower inimitability. Enterprise-SaaS, on the other hand, refers to software which is more

complex and which may require support, involving integration with customers' existing IT systems [15].

Several studies have focused on the adoption of SaaS (e.g. [11, 18]). From the findings of Wu et al. [18], it appears that the adoption of SaaS is more related to strategic benefits than to economic benefits. In addition, SaaS adoption is more related to subjective risks than to technical risks. Benlian and Hess [11] focused on IT executives' perceptions of the opportunities and risks in SaaS adoption. They found that security threats were the most dominating factor in risk perception, whereas opportunities related to the cost advantages of SaaS were the strongest driver for SaaS adoption. Fan et al. [19] studied short-term and long-term competition between SaaS providers and traditional software providers. According to their findings, SaaS can reduce price competition and facilitate differentiation from traditional software providers. However, SaaS providers may incur significant operating costs due to the fact that they have to invest in service and system capacity in order to guarantee availability of the service [19].

### B. Pricing models in software business

Software pricing may be based on (i) assessment base pricing, (ii) price discrimination, and (iii) price bundling, and combinations of these. In assessment base pricing, the pricing models can be divided into *usage-dependent* pricing, *user-based* pricing, and a combination of the two [8, 20]. In the usage-dependent model, the fee is related to the number of transactions, memory requirements, and so on. In the user-based pricing model, customers pay a fixed fee for unlimited use (see more detail [20]).

Price discrimination refers to a pricing model in which the same product is offered to different customers at different prices [16, 17, 20, 21]. The price discrimination model can be divided into first-degree, second-degree, and third-degree price discrimination. In first-degree price discrimination, the price is set according to the customer's ability/willingness to pay [21]. Second-degree price discrimination refers to the quantity, time (season or duration), version, or market segmentation that is used to set the price (cf. [20]). In third-degree price discrimination, customers are charged according to identifiable sectors, such as occupation, location, or age [22]. In addition, software firms may use combinations of different degrees of price discrimination (cf. [20]).

Price bundling refers to a pricing model that includes several components, for example software products and/or services packaged together for a fixed price [23]. By bundling different products and/or services together, software providers can make their offering more attractive to customers than by selling the same products separately [23]. Price bundling can be divided into *offer*, *product*, *degree of integration*, and *price level* aspects [20].

### C. The economics of renting vs. purchasing

In economics literature, authors have frequently focused on the benefits of renting vs. purchasing in monopolistic situations in the market [5, 6]. According to Flath [6, p. 247], renting can be defined as "a contractual arrangement for trading the rights to temporary use of an object, but not the right to all possible future use." Thus, in a rental agreement, a customer does not get the full ownership rights over the object rented, as distinct from ownership following purchase. However, there are always trade-offs between the benefits of full ownership and those of "partial ownership" – i.e. renting.

From the customer's point of view, these benefits are related to the characteristics of the product and the time period needed for usage of the product [5, 6]. In other words "the shorter is one's expected tenure of use of a good, the greater are the transacting cost gains to his leasing it rather than purchasing it outright" [5, p. 249]. According to Choudhary et al. [7] the reasons why a customer may rent software in preference to purchasing it are as follows: (i) the software is for use in a short-term project, (ii) the customer may simply want to gain experience of using the software, (iii) the customer wants to test and evaluate the usability of the software, or (iv) the customer wants to avoid negative network externality.

From the software provider's perspective, renting decreases transaction costs related to identifying, assuring, and maintaining quality [3, 7, 9]. Renting can also increase the positive network externality effect [24], owing to the lower initial costs for customers compared to purchasing. The low costs increase the number of customers, and consequently increase the information available in the market regarding the product. Overall, this decreases customers' search costs [7].

## III. METHODOLOGY

The research method selected for this study covered a real-life environment in which there was a decision-making process related to revenue and pricing models. It was important that the method should cover human actions, enable an in-depth investigation of the complex phenomena at work, and capture cause-and-effect relationships. With all this in mind, this study applied a multiple case study methodology similar to the approaches presented by Eisenhardt [25] and Yin [26].

The research setting for this study consisted of four software firms that acted as SaaS providers. Multiple sources of information were used to gather data on each case firm. The main form of data collection consisted of in-depth interviews. Altogether, 6–10 interviews per firm were conducted, each lasting 45–90 minutes. Thus, altogether 32 semi-structured open-ended interviews were carried out for this study (see Table 1). All the interviews were recorded and transcribed them verbatim, using a word processing program. Thereafter, the complete transcripts were sent back to the interviewees for review. In addition to the face-to-face interviews, telephone and e-

mail communication was used to collect further information, and to clarify inconsistent issues if necessary. By comparing the interview data with other information gathered on the case firms, triangulation of the information was conducted [27].

TABLE 1.  
OVERVIEW OF THE DATA SOURCES

Firm	Informant title	Number of interviews	Total number of interviews
Firm A	CEO	1	8
	Vice President	4	
	Vice President	3	
Firm B	CEO	6	10
	Chairman	1	
	Vice President	1	
	Executive Director	1	
	General Manager	1	
Firm C	CEO	2	6
	Accounts Manager	2	
	Sales Manager	2	
Firm D	CEO	2	8
	Art Director	2	
	CTO	1	
	COO	1	
	Head of Sales	1	
	Sales Engineer	1	

The method utilized in the data analysis was content analysis. The analysis of the case data consisted of three concurrent flows of activity [27]: (i) data reduction, (ii) data displays, (iii) conclusion-drawing/verification. In the data reduction phase, the data were given focus and simplified through compilation of a detailed case history of each firm. On the basis of the interviews and other material collected from the case firms, tables were used to identify and categorize the unique patterns of each case. In the data display phase, the relevant data drawn from the case descriptions were arranged in new tables. These tables included direct citations from the interview data that were later used in the findings section. In the phase of conclusion-drawing and verification, the cross-case aspects that appeared to have significance for this study were identified.

#### IV. WITHIN-CASE ANALYSIS

**Firm A** develops planning and optimization software for telecom operators. The firm, established in 1998, acted purely as a service firm until 2009, when it started to develop its own software product. The idea of bringing the product directly to the cloud model and of having multi-instance architecture was based on the complexity of existing products in the market. These products required purchase of a costly license and installation to a workstation, with the software then becoming accessible only through a particular computer. Thus, by the pure-SaaS model, all the participants in a project gained easy and instant access to the software wherever they were located physically. From the outset, the software

architecture was designed to be compatible with both software renting and software licensing, depending on the customers' needs. The CEO explained this as follows: *"We have different ways to offer the product to customers. We can sell it as SaaS, using software renting. The second option is to sell it via a traditional software license, if someone prefers that."*

The SaaS model made software renting easier and enabled customers to rent the software for short-term projects. Firm A prefers software renting on the grounds that it brings cost savings by decreasing production costs. These cost savings are related in particular to the traveling costs for installation, implementation, maintenance, and after-sales support within customers' premises.

However, not all customers are willing to rent the software through the public cloud, due to security concerns or a low readiness for new technologies. For these customers, Firm A makes available a traditional licensing model, in which the customer buys a traditional license for its internal data center. In many cases, a customer had its own IT infrastructure allowing the software to be implemented in-house, decreasing the need for rent the software through SaaS model. Another reason to sell a traditional license for internal use was a lack of network connections, in certain cases where a customer was in the process of building a new telecom network. The managers of Firm A felt also that the rental model also made the software attractive for the customer as it was possible to have more flexible pricing models than in software licensing.

The pricing was depending on the customer's needs for usage of the software. Hence, Firm A set the price for the software according to the number of users within the organization (concurrent user-based pricing) and the length of the agreement (time-based price discrimination). The firm tried to promote its rental model by pricing the rental model to be economic choice than the licensing model. Agreements were commonly made for at least three months, but this was negotiated with each customer separately. The vice president explained this as follows: *"We try to avoid traditional licensing by using a pricing model that makes the cloud model and software renting much cheaper for the customer"*

**Firm B**, established in 2000, provides interactive gaming platforms and games-on-demand services, using multi-tenant software architecture. The firm licenses game content from game developers and integrates these games within its gaming platform. The network operators operate the platform and deliver games to players' set-top boxes or to PCs through their broadband network, in the form of pure-SaaS. The revenue model is based on software renting, with a variety of payment options for consumers. These options include (i) a subscription based on named user-based pricing for a particular game for 24 hours, (ii) a monthly subscription for all the games available in the service, and (iii) a subscription for different games packages for a certain time period. The CEO commented this as follows: *"This revenue model allows us to have very interesting pricing strategies."*

*Instead of buying the game from a store, this makes it possible to rent the game for a day ...or the game can be part of a games package that can be used for a month."*

Thus, in addition to named user-based pricing, Firm B uses time-based price discrimination and price bundling. This model provides flexible pricing options for consumers, and these increase the attractiveness of the service compared to the traditional way of buying games from a store. However, the final revenue is shared with the partners in the value chain. The CEO explained the situation as follows: *"It is based on revenue sharing between three players, the content owner, a portal, and ourselves –all sharing the content revenue. If the content is [equivalent to] five dollars, we share it so that there are two dollars for the portal, and two dollars for the content owner, and one for us, or something like that."*

Because the content is delivered through network operators to game players, the network operator determines the revenue model used. As the CEO put it: *"In practice, we could also use a pay-per-use model, for instance if a player plays the game let's say 15 minutes, we will charge him a price unit that corresponds to 15 minutes. However, network operators have become accustomed to using rental models and that is why we have to use one too."*

**Firm C** was launched in 2008 as a spin-off from a software service firm that provides consulting services related to identity management. Currently, Firm C develops enterprise-SaaS for access rights management. The software can be used in many different types of organizations, including financial institutions and government organizations. From the outset, the software has been available for customers via either a rental model or a traditional software license. The CEO commented this as follows: *"We have two revenue models – SaaS where there is a price per user per month, or else a customer can buy a traditional license, which is like a concern license. By using the concern license the customer has free usage of the software within the organization."*

The software architecture is multi-tenant for both revenue models. In line with Firm A, this approach was adopted because both models have been asked for by customers. The sales manager explained this as follows: *"We like to have both options available because both are sought by customers. We don't want to turn this into a restrictive issue... Technically there are no differences, so if the customer says that the product cannot be allowed out of their internal data center, then we will install it there."*

However, although most of the customers bought the license, the firm was moving towards a rental model on the grounds that an SaaS rental model decreased production costs as the implementation process is faster as the customer can take the existing system into use over the Internet. For some customers it was of no importance where the software was installed physically, whereas some customers wanted to use it only in their internal data center, due to security concerns or a lack of confidence in

the reliability of the public cloud. In addition, customers with their own IT department were more likely to prefer a traditional license, whereas customers who were accustomed to outsourcing their IT services favored software renting. Furthermore, some customers had an IT policy that did not allow installation of any software outside their own internal data center.

Under the rental model, there is a monthly fee, which is dependent on the estimated number of users within the organization (concurrent user-based pricing). Interestingly, in the case of Firm C, the length of the agreement did not impact on the total rental fee, since that firm's software required labor-intensive implementation work, and this work increased switching costs. The CEO explained this as follows: *"The length of the agreement does not impact on the rental fee. We do not have low switching costs because the amount of project work needed to get the system to work is so great. So I don't see that switching costs are a problem in our case."*

**Firm D** was established in 2006 as a spin-off from a larger software firm. It develops 3D modeling software for furniture manufacturers and furniture retailers as enterprise-SaaS. When the firm was established, the first idea was to sell the product for each customer's internal data center. This idea was based on the model that the founders had used in their previous firm. However, quite soon they realized that the most of their customers did not care where the software was running so long as it worked properly. The CEO commented as follows:

*"We first thought of installing the software in each customer's server. However, within the first six months we realized that this does not work with smaller customers who do not have servers in the firm. It became clear that 90 percent of our customers wanted to use this through a public cloud"*

Hence, Firm D currently uses only a rental model with multi-instance architecture. The multi-instance approach makes it possible to add customers' own furniture elements to the software within the meta-data layer. In addition, because the software of Firm D does not handle sensitive data, security concerns were not a critical issue. The initial rental agreement was for three years, and thereafter a customer could renew it for a fixed period. As the software required substantial implementation work, having a three-year subscription agreement ensured that Firm D could cover the development of the software.

The managers described that software renting made it possible to use flexible pricing for the software. In addition to concurrent user-based pricing, the rental fee depended on the functionalities (version-based price discrimination) and the number of furniture elements included (quantity-based price discrimination) in the software for the rental period. As the COO put it: *"The rental fee is based on the version that a customer selects and the amount of content that the customer wants to have in the software. For instance there might be a sofa including 50 elements or 40 sofas including something like 500 different elements. We have certain stages according to which we price the number of elements. So*

*we don't charge according to the number of users, we charge according to the functionalities and the number of elements included."*

## V. CROSS-CASE ANALYSIS

The findings here complement and expand on earlier studies on software renting and related pricing models. Firstly, the case findings show that, in both revenue models, a great variety of software pricing models are available, and firms can use these individually or in combination. In a way similar to licensing options, software renting enabled flexible pricing (firms A, B, and C) based on the number of users, the functionalities used, and so on. In addition, software renting made it possible to change the pricing according to the named users (Firm B) or concurrent users (firms A, C, and D) during or after the rental period (based on the rental agreement). Low initial costs in software renting enabled to expand the customer base from large-sized customers toward small and medium-sized customers (Firm D). Furthermore, renting made it possible to offer the software at reasonable price if a customer needed it only for a short-term usage. In firms A, C, and D the pricing was based on concurrent user assessment, taking into account the need to protect their business against increasing maintenance costs, whereas Firm B used named user assessment and price bundling. Thus, the number of concurrent users increased the price, as the number of users correlated with the capacity and costs required for data storage and computing power. A larger number of users also made installation and customer support more complex and time consuming. These finding complements earlier studies [8] that have studied the selection of a proper pricing model only between binary models. Altogether, the findings reveal that software firms can use flexible revenue and pricing strategies to make the service more attractive for customers.

Secondly, the rental model through the public cloud decreased the production costs of case firms A and C. The case firms knew that the customers were using the same version of the software, and by means of the public cloud, the case firms were able to bring updates that were visible to all their customers immediately. The technical features of the model consistently brought cost savings to the case firms and made it possible to offer the software at a lower price. The factor of lower production costs gives empirical support to the model of competition between SaaS providers and traditional software providers [19]. However, differentiation was based not only on lower implementation costs – as argued by Fan et al. [19] – but also on faster and more cost-effective delivery, maintenance, and after-sales services within the SaaS model. In contrast to previous studies [7, 9] that have indicated lower transaction costs in the case of software renting, none of the case firms mentioned this as an advantage. This was related to the fact that in software renting, the negotiation costs and contract monitoring costs related to renting offset the benefits of other

transaction cost advantages (advantages which would include the lower costs of identifying, assuring, and maintaining quality; see [7, 9]).

Thirdly, firms A and C were able to use both revenue models, using the same software architecture for software licensing and renting. Hence, the usage of parallel revenue models does not include any extra work on the technical level, so long as the software architecture has been developed to suit both models. Based on the findings, it also seems that the SaaS architecture does not impact on revenue or pricing model used in the market. In contrast, the SaaS business archetype seems to impact to the selection of software pricing. Pure-SaaS providers (Firms A and B) seem to use time-based price discrimination to protect their business against low switching costs. Hence, the fee for a short-term rent agreement was higher than for a long-term contract. In contrast, enterprise-SaaS providers (Firms C and D) seem to protect their business against low switching costs by using separate pricing for the implementation work. The implementation of their software required so much labor-intensive project work that customers' switching costs increased, offsetting the benefits of short-term renting.

Fourthly, data security concerns played a significant role in the adoption of SaaS (firms A and C) – an aspect examined by earlier studies [3, 11]. In contrast, firms B and D, who used only software renting, did not experience problems with security concerns. This was mainly because of the software offerings (video games and 3D modeling software), which were aimed at activities in which there were no security issues for the customer. The findings reported here expand on earlier studies by demonstrating how data security concerns are related to the revenue model of a firm. The findings also demonstrate how SaaS providers may be forced to offer their product under license for customers with high importance. Otherwise, the firm will simply lose these customers.

Finally, according to our findings, the customer's IT capability has a strong impact on deciding whether to buy or to rent the software (firms A and C). If customers had already invested in IT infrastructure and their own IT personnel, they saw traditional software licensing as a more attractive choice. This was mainly because the benefits of software renting became less significant, and operating the software in-house increased the feeling of trust. This is an aspect neglected in previous studies on software renting [3, 7].

## VI. CONCLUSIONS

This study incorporates theoretical insights from economic literature to enrich and extend IS research on software firms revenue and pricing models. Earlier studies on software renting have focused on the economic benefits of renting [7], the impact of renting on the quality of the software [3], the profitability of renting [9], and software pricing [8, 20]. This study extends and complements these earlier studies by revealing how

software firms use revenue and pricing models to make their software attractive for customers.

It can be concluded that customers or delivery channels are the main driving force to the selection of the most appropriate pricing and revenue model in the market. Software firms seem to favor renting as it brought several benefits such as flexible pricing and lower production costs. In addition software renting enabled firms to protect their business against low switching costs and made the software also available to smaller customers that were not able to pay high initial costs related to the licensing model. However, traditional software licensing was also needed in some cases. Customers who had data security concerns or who had already invested in IT infrastructure and IT personnel were more willing to invest to the traditional software license and operate the software in-house.

#### ACKNOWLEDGEMENTS

The research reported here was carried out within the framework of the Digital Services research project financed by TEKES -the Finnish Founding Agency for Technology and Innovation. In addition, the Foundation for Economic Education in Finland (Liikesivistysrahasto) provided financial support for this study.

#### REFERENCES

- [1] Luftman, J. & Zadeh, H.S. Key information technology and management issues 2010-11: an international study. *Journal of Information Technology* 26(3): 193-204, 2011.
- [2] Ferrante, D. Software Licensing Models: What's Out There? *IT Professional* 8(6): 24-29, 2006.
- [3] Choudhary, V. Comparison of Software Quality Under Perpetual Licensing and Software as a Service. *Journal of Management Information Systems* 24(2): 141-165, 2007.
- [4] Leavitt, N. Is Cloud Computing Really Ready for Prime Time? *Computer* 42(1): 15-20, 2009.
- [5] Bulow, J.I. (1982) Durable-Goods Monopolists. *Journal of Political Economy* 90(2): 314-332.
- [6] Flath, D. The Economics of Short-term Leasing. *Economic Inquiry* 18(2): 247-259, 1980.
- [7] Choudhary, V. Tomak, K. & Chaturvedi, A. Economic Benefits of Renting Software. *Journal of Organizational Computing and Electronic Commerce* 8(4): 277-305, 1998.
- [8] Sundararajan, A. Nonlinear Pricing of Information Goods. *Management Science* 50(12): 1660-1673, 2004.
- [9] Varian, H. Buying, Sharing and Renting Information Goods. *The Journal of Industrial Economics* 48(4): 473-488, 2000.
- [10] Armbrust et al. A view of cloud computing. *Communication of the ACM* 53(4): 50-58, 2010.
- [11] Benlian, A. & Hess, T. Opportunities and risks of software-as-a-service: Findings from a survey of IT executives. *Decision Support Systems* 53(1): 232-246, 2011.
- [12] Ju, J., Wang, Y., Fu, J. & Lin, Z. Research on Key Technology in SaaS. *ICICCI conference*, 384 -387, 2010.
- [13] Sun, W., Zhang, X., Guo, C. J., Sun, P. & Su, H. Software as a service: Configuration and customization perspectives. *IEEE Congress on Services 2008*, pp. 18-25, 2008.
- [14] Benlian, A., Hess, T. & Buxmann, P. Drivers of SaaS-adoption - an empirical study of different application types. *Business & Information Systems Engineering* 1(5): 357-369, 2009.
- [15] Luoma, E., Rönkkö, M. & Tyrväinen, P. Current Software-as-a-Service Business Models: Evidence from Finland. In M.A. Cusumano, B. Iyer & N. Venkatraman (Eds.) *Software Business*, 114, 181-194, 2012.
- [16] Laatikainen, G., Ojala, A. & Mazhelis, O. Cloud services pricing models. *Proceedings of 4<sup>th</sup> International Conference on Software Business (ICSOB 2013)*, Potsdam, Germany, June 11-14.2013.
- [17] Laatikainen, G. & Ojala, A. SaaS architecture and pricing models. *Proceedings of 11<sup>th</sup> IEEE International Conference on Service Computing (IEEE SCC 2014)*. Anchorage, Alaska, USA. June 27 – July 2.2014.
- [18] Wu, W-W., Lan, L.W. & Lee, Y-T. Exploring decisive factors affecting an organization's SaaS adoption: A case study. *International Journal of Information Management* 31(6): 556-563, 2011.
- [19] Fan, M., Kumar, S. & Whinstor, A.B. Short-term and long-term competition between providers of shrink-wrap software and software as a service. *European Journal of Operational Research* 196(2): 661-671, 2009.
- [20] Lehmann, S. & Buxmann, P. Pricing Strategies of Software Vendors. *Business & Information Systems Engineering* 1(6): 452-462, 2009.
- [21] Hinz, O., Hann, I. & Spann, M. Price discrimination in e-commerce? An examination of dynamic pricing in name-your-own price markets. *MIS Quarterly* 35(1): 81-98, 2011.
- [22] Adachi, T. Third-Degree Price Discrimination, Consumption Externalities and Social Welfare. *Economica* 72(285): 171-178, 2005.
- [23] Bakos, Y. & Brynjolfsson, E. Bundling Information Goods: Pricing, Profits, and Efficiency. *Management Science* 45(12): 1613-1630, 1999.
- [24] Liebowitz, S.J. & Margolis, S.E. Network Externality: An Uncommon Tragedy. *The Journal of Economic Perspectives* 8(2): 133-150, 1994.
- [25] Eisenhardt, K.M. Building theories from case study research. *Academy of Management Review* 14(4): 532-550, 1989.
- [26] Yin, R.K. *Case study research: Design and methods*. CA: SAGE Publications, 2009.
- [27] Miles, M.B. & Huberman A.M. *Qualitative Data Analysis: An Expanded Sourcebook*. California: Sage Publications, 1994.