



Interoperability of Software: Demand and Solutions

Knut Blind

Faculty of Economics and Management, Berlin University of Technology, Germany
and Fraunhofer Institute for Systems and Innovation Research, Karlsruhe, Germany
(eMail: Knut.Blind@TU-Berlin.de)

Abstract This paper presents empirical evidence on the demand for interoperability among German software companies. We test a set of hypotheses in the empirical analyses. The findings confirm that the interoperability with the software of their customers is most important. The multivariate analysis shows that the demand for interoperability increases with company size, confirming the concept of network externalities, especially between users. In addition, for companies trying to achieve lead-time advantage, interoperability is also of high importance to secure the installed user base. Finally, the demand for interoperability depends on the characteristics of specific software products. The most important instrument to achieve interoperability is the disclosure of interfaces, the second preference is the use of standardised architectures

Keywords software, interoperability, interfaces, standards

1. Introduction

From an economic perspective, it is well known in theory (Shy 2001) that the greater the interoperability of two complementary components, like software and hardware, the greater too is the diversity of complementary inputs, like further software products, which are at the user's disposal. However, the empirical evidence on the interrelation between interoperability and variety in the software market is rather limited (one exception is David and Greenstein 1990).

Another question, which is not extensively dealt with either in the theoretical literature or in empirical studies, is the demand for interoperable solutions. Since we observe an increasing complexity of the software development process, various dimensions regarding interoperability for companies producing software emerge. First, a software developing company

thinks immediately about the need to produce software which is interoperable with the software its customers use, which they may have even developed themselves. However, software developing increasingly takes place within a complex value chain, which may also require integrating software from other supplying companies. This will be facilitated, if interoperability is realised with the supply side. The demand for own software depends also on its relation to other products on the market. Here we have to distinguish two dimensions. First, the interoperability of own software with complementary software products, which are used by potential customers, increases the demand and the value of the own product. Second, interoperability with the software products of competitors is ambivalent. Dominant players have little interest in interoperability with competing products, because price competition especially will drive down their profits (Besen, Farrell 1994). Small companies or companies entering the market should have a stronger interest to provide products which are interoperable with those of the incumbents or the dominant players, in order to use their so-called installed base of users. The need to join installed bases of users is higher for software, which requires both high development costs and which is characterised by strong positive network externalities. In total, whether companies desire interoperability with the software products of competing products depends on several factors.

The empirical data allows us to demonstrate different preferences for the interoperability in the four dimensions discussed above. Furthermore, we will estimate four demand functions for interoperability depending both on firm and software characteristics.

In a second step, we have to discuss how interoperability is achieved. In mainstream economic literature, interoperability is either not used at all (Shy 2001) or is used more or less synonymously with compatibility. And compatibility is achieved simply via standardisation (compare for an overview of empirical studies Blind 2004). In economic literature, we find also the option to develop gateways between technologies (David, Bunn 1988). However, gateway technologies are more appropriate to achieve interoperability or compatibility between hardware components in order to generate mostly direct network externalities, e.g. in the case of telecommunication. In software, gateway solutions do not play a significant role.

In our study we distinguish between three different standardisation strategies. First, companies may use in their software the specifications of standardised sector-wide architectures, which are open and transparent. Second, they may orient their software towards the de facto standards of the market leader. In contrast to the two passive standardisation strategies, software companies may try to get their own specifications accepted as de facto standards. A step towards interoperability, which is a pre-stage of standardisation, is the disclosure of interfaces or even of the complete

source code. Both strategies are closely related to the promotion of the specifications towards de facto standards. Finally, the problem of interoperability can already be tackled at a very early stage by cooperating in the joint development of software or at least of common interfaces. In the same way as for the demand for interoperability in the four dimensions discussed above, we will develop hypotheses that will determine the demand for these instruments, which includes also for which interoperability dimension they are used.

The remainder of the paper is organised as follows. In the next section, we present the data source and the main characteristics of the companies surveyed. Chapter three presents descriptive statistics regarding the demand for the four dimensions of interoperability. Besides the descriptive statistics, we present the results of multivariate ordered Probit models, which allow us to determine the driving forces for the demand for interoperability. In chapter four, we present descriptive statistics regarding the use of various instruments to achieve interoperability. The paper concludes with a summary of the results trying to identify some stylised facts regarding the interoperability of software.

2. The database and major characteristics of the sample

2.1 The database

The data for testing the hypotheses regarding the demand for interoperability and the means to achieve it originates from a survey of German software developing companies performed in the year 2001 (Blind et al. 2003; see English version complemented by case studies in Blind et al. 2005). A basic hypothesis of a former empirical study (Stahl et al. 2000) presupposes significant differences between enterprises whose main corporate aim is to develop and sell software (primary sector), and enterprises of the so-called secondary sector, which besides a traditional main business also produce software for their 'hardware'. Software, which is irrevocably integrated in hardware components and can only function together with them, such as e.g. specific control software in mechanical engineering or vehicle construction, we refer to as embedded software. For this reason the presentation of the descriptive statistics is differentiated according to these two groups.

In the survey, those enterprises belong to the primary sector whose corporate objective is the development of software, according to the generally accepted classification NACE (NACE 72.202). They are supplemented by a number of independent software developers, because these micro-companies play a considerable role in the software area (Stahl et al. 2000). The industries of the so-called secondary sector were vehicle construction

(NACE 34), electro-technology (NACE 30-32), telecommunications (NACE 64) and mechanical engineering (NACE 29). The survey was conducted in May and June 2001 by approaching software companies or companies developing also software via an online questionnaire. In total 149 questionnaires from companies of the primary sector, 39 questionnaires of independent software developers and 68 questionnaires from companies of the secondary were available for the analysis.

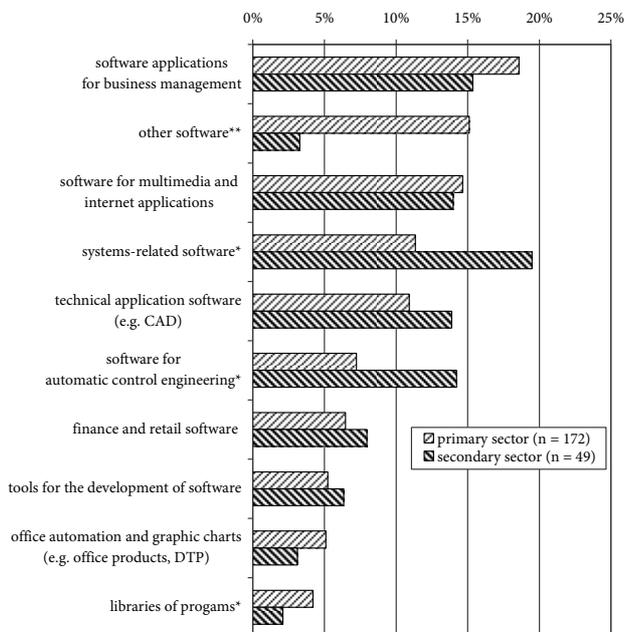
2.2 Types of software developments and their significance

Since the need for and the instruments to achieve interoperability is depending on the type of software being developed, the firms were first of all asked how their turnover was distributed among various types of software. A first dimension hereby is the independence of the developed product, which ranges from the so-called stand-alone solutions to complete integration in other software, respectively hardware. The average turnover share with independent software products amounts to somewhat over 58% in the primary sector, in the secondary sector to somewhat more than 35%. For the turnover share of so-called 'embedded software', that is, software which is an integral part of hardware and only functions in conjunction with it, this ratio is reversed, as expected.

In the primary sector, the average share of embedded software amounts to only 7%, while it accounts for nearly 36% in the secondary sector and thus corresponds mainly to the share of independent software. This distribution corresponds to the claims about own production of hardware: 85% of the enterprises in the secondary sector and 44% of the firms in the primary sector develop, respectively also produce hardware. These differentiations according to primary and secondary sector are both statistically significant. These distributions mean that the particular problems of embedded software do not play an essential role for the primary sector, whereas for the secondary sector not only the specifics of independent software, but also the idiosyncrasies of embedded software are relevant. Furthermore, in both sectors hardware producers also increasingly produce embedded software.

As the demand for and the means to achieve interoperability will depend on the functionality of their products, the sample was asked about the various application areas of their software products (Figure 1). It appears that the secondary sector has the largest share of turnover with systems-related software, followed by approximately equal shares of user software from the areas business management, technical applications, automatic control engineering and multi-media. In the primary sector, on the other hand, the user software plays a greater role (with business administration software in the lead), systems-related software takes third

Figure 1 Turnover shares with software of different functionality



Source Blind et al. (2005)

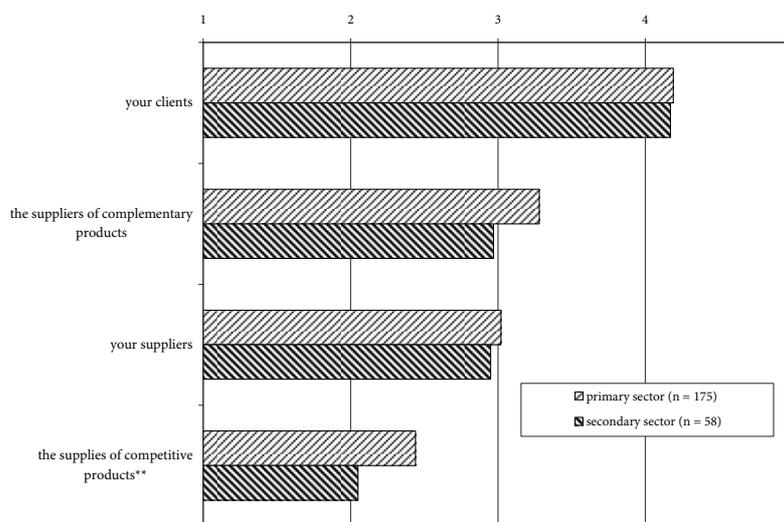
place here with only 11%.

A final differentiation among the firms regarding their products distinguishes according to what extent they develop their products as individually tailored software solutions for single customers, respectively small groups of customers. The underlying assumption is that the patenting of products for the mass market is of greater significance than for custom-made articles. The results show that in the secondary sector most often small series are developed for certain customer groups, whereas the primary sector is more strongly polarised and clearly more than a third of their turnover stems either from client-specific, customised developments or from products for the mass market.

3. Demand for interoperability

In a first step, we present the results relating to the hypotheses about the importance of interoperability with suppliers or customers, or with competitors. One idiosyncrasy of software development in contrast with other products is the interoperability necessary between systems and applications or between various applications. As argued above, four dimensions

Figure 2 Significance of interoperability with software of various actor groups (1 = very low, 5 = very high)



Source Blind et al. (2005)

must be differentiated regarding interoperability. Figure 2 shows, that the interoperability with customer software is by far the most important criterion for both sectors. The interoperability with supplier products is of medium importance and is approximately as significant as the interoperability with products of complementary suppliers. The significantly higher figure of the primary sector in the interoperability with competitive products reveals the importance attributed to the functional compatibility of the own product with others on the market. On the other hand, this strategic motive is not pronounced in the secondary sector.

In order to test the hypotheses about the factors, i.e. company or product characteristics, influencing the demand for interoperability, we apply the following approach. Multivariate ordered Probit analyses allow us to determine the influence of single variables on the assessment of the importance of interoperability, controlling for all explanatory factors simultaneously. The general ordered Probit model equation, which we apply for the four different categories of interoperability (Figure 2), is specified as follows:

$$\text{OProbit(Importance of Interop)} = f(\text{Company Characteristics}; \\ \text{Company Strategy}; \\ \text{Product Characteristics})$$

Table 1

The company characteristics we have included in our various models are:

- company size measured by the number of employees (size)
- R&D intensity as percentage of R&D expenditures of turnover (R&D intensity)
- dummy variables for companies of the primary sector (primary sector dummy) or independent software developers (independent developer dummy) with the companies of the secondary sector as base category.

The company strategy is characterised as:

- the self-assessed importance to achieve lead-time advantage, e.g. introducing new generations of products before competitors (lead time).

The product characteristics are defined according to the differentiation in Figure 1.

In the following, we report the most important results (see Table 2.1 for the detailed results) in order to elaborate some stylised facts on the relevance of interoperability.

First, we observe that the importance of interoperability with customers, suppliers and competitors providing complementary products increases with company size. However, the importance of interoperability regarding competitors supplying substitutive products is not significantly related to company size. This result indicates that the demand for interoperability increases with company size due to economies of scale, which require larger firms to serve a larger group of customers. Regarding the importance of interoperability with customers, we observe a U-shape, i.e. very small companies produce and sell very customer-specified products, whereas very large ones have to exploit network externalities between user networks. Therefore, we find also the strong size effect in the equation to explain the demand for interoperability with suppliers of complementary products (Succi et al. 1998), but not for the demand for interoperability with suppliers of competitive products.

The R&D intensity does not explain the importance of interoperability regarding customers and competitors providing complementary products. However, it turns out that companies with higher R&D intensity have a lower demand for interoperability respectively their suppliers and their competitors supplying substitutive products. This result provides some support for the trade-off between R&D intensity and interoperability. Companies with lower R&D activities try obviously to benefit either from the input of their suppliers or from the network effects of their direct competitors.

The independent software developers reveal a higher demand for interoperability with products of suppliers both of complementary and substitutive products. This observation is in line with the traditional picture and philosophy of the Open Source community.

Companies trying to achieve lead-time advantage respective to their competitors have a higher demand for all four types of interoperability. The compatibility with the existing installed base is obviously very crucial for those companies trying to be at the leading edge.

Finally, the characteristics of software products have an influence on the importance of interoperability. We find the following product-related results. Those companies developing computer games and tools for software development have a lower demand for interoperability. In contrast, companies producing office products have a higher demand for interoperability, respectively their customers. Companies developing business administration software have lower demand for interoperability with suppliers of substitutive products, whereas companies providing finance and retail software have higher demand for interoperability with suppliers of substitutive products.

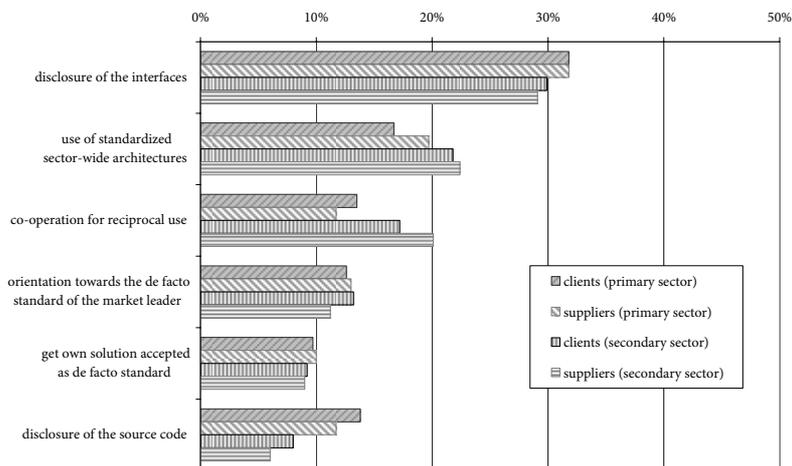
4. Means to achieve interoperability

Since there are different options to achieve interoperability, we must ask first of all how interoperability can be established at all. In this context, it makes sense to differentiate between the vertical level of client and supplier relationships (Figures 3) and the horizontal level of competing and complementary products (Figures 4).

By far the most frequently mentioned instrument for all four dimensions is the disclosure of interfaces, followed by the use of standardised sector architectures, which is obviously somewhat more widespread within the secondary sector. The same relation applies to contractual co-operations, which is accorded slightly less importance. On the other hand, code disclosure plays rather a subordinate role, above all in the secondary sector. It is surprising that, by comparison to the orientation to the de facto standard of the market leader, only slightly fewer firms attempt to have their own specifications accepted as a de facto standard.

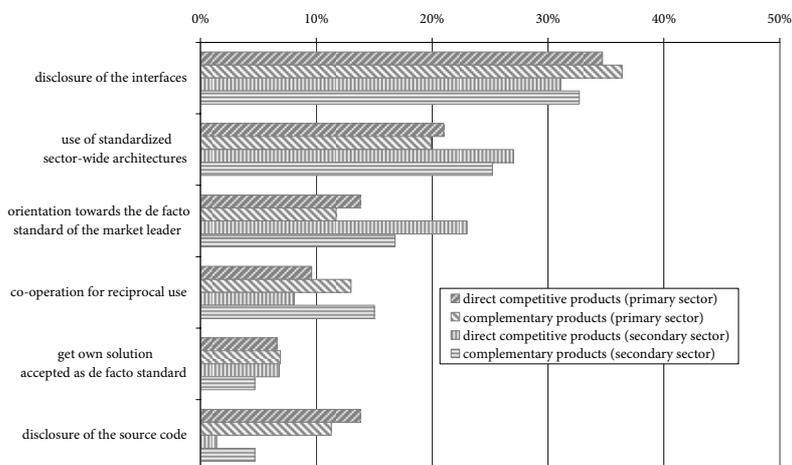
If one leaves the vertical level of client and supplier relationships and turns to the direct competitive relations divided according to suppliers of complementary and competitive products, similar structures emerge. On the one hand, the disclosure of relevant interfaces is the preferred strategy of around one third of all enterprises, not only for complementary, but also for competitive products. Further, a quarter of the firms in the secondary sector use standardised sector architectures. Only a fifth does so in the primary sector.

Figure 3 Securing interoperability to clients and suppliers



Source Blind et al. (2005)

Figure 4 Securing interoperability with competitive and complementary products



Source Blind et al. (2005)

It is remarkable that regarding the direct competitive products above all in the secondary sector, an orientation to the sector standard takes place, while direct legally contracted co-operations are entered into especially with suppliers of complementary products. The enterprises obviously have greater difficulties cooperating with suppliers of direct competitive

products on a contractual basis, as in certain circumstances conflicts can be caused in competition law by this.

In the case of complementary products, these and other risks are less, in contrast the incentives for direct co-operations are obviously more pronounced. The disclosure of own code does not play a role in either sector finally and within the primary sector is only practiced by the independent developers.

5. Summary

This paper attempts to present some first insights on the need for interoperability in the software sector and on the means to achieve the different dimensions of interoperability. The basis for the empirical analysis was a sample of German software companies. We have analysed the factors influencing the demand for interoperability, which is certainly also reflected in the supply of their software products. It turns out that with company size the demand for interoperability increases, although we observe also the pattern of an inverted U-shape. Furthermore, there are indications that R&D intensive has a lower need for interoperability, which gives reason to assume that there is some trade-off between innovation and interoperability. However, those companies trying to be ahead of competitors have a strong preference for interoperability in order not to lose their customer base. Finally, those companies involved in open source activities are definitely willing to secure the interoperability of their products with those of other suppliers. These first stylised facts on the interoperability of software represent a challenge for developing theoretical models addressing them.

In a separate chapter, we have presented the use of different means to achieve interoperability. Further investigations require the development of a more comprehensive set of hypotheses and theoretical model, which combine the demand for interoperability with the means to achieve interoperability.

Based on the insights of these more comprehensive approaches, policy makers can be informed how to deal with the future challenges to provide adequate frameworks for interoperable solutions especially for e-business and e-government.

Acknowledgements

The author thanks for the financial contribution for the NO-REST project (507626) by the European Commission within the 6th Framework Programme. The paper was developed within the NO-REST project. The data was generated in a study funded by the German Ministry for

Economic Affairs. A previous version of the paper was published in the proceedings of the INTEROP workshop organised in Geneva. Comments by an anonymous reviewer contributed to a significant improvement of this paper. Remaining shortcomings are the responsibility of the author.

References

- Besen, S.M. and Farrell, J. (1994), Choosing how to compete: Strategies and tactics in standardization, *Journal of Economic Perspectives* 8: 117 – 131.
- Blind, K. (2004), *The Economics of Standards: Theory, Evidence, Policy*, Edward Elgar.
- Blind, K. et al. (2003), *Software-Patente: Eine empirische Analyse aus ökonomischer und juristischer Perspektive*, Physica-Verlag.
- Blind, K. et al. (2005), *Software Patents: Empirical Evidence and Policy Implications*, Edward Elgar.
- David, P.A. and Greenstein, S. (1990), The economics of compatibility standards: An introduction to recent research, *Economics of Innovation and New Technology* 1: 3 – 41.
- David, P.A. and Bunn, J.A. (1988), The economics of gateway technologies and network evolution, *Information Economics and Policy* 3: 165 – 202.
- Shy, O. (2001), *The Economics of Network Industries*, Cambridge University Press.
- Stahl, P. et al. (2000), *Analyse und Evaluation der Software-Entwicklung in Deutschland. Eine Studie für das Bundesministerium für Bildung und Forschung, GfK Marktforschung (Nürnberg), ISI (Karlsruhe), IESE (Kaiserslautern)*.
- Succi, G. and Valerio, A. et al. (1998), Compatibility, standards and software production, *StandardView* 6: 140 – 146.