Product Variety and Firm Survival in the Microcomputer Software Industry

Tom Cottrell

And

Barrie R. Nault

University of Calgary
Haskayne School of Business
2500 University Drive NW
Calgary, Alberta, Canada
T2N 1N4

Cottrell: cottrell@ucalgary.ca; (403) 220-4477; (403) 282-0095 (fax).
Nault: nault@ucalgary.ca; (403) 220-2742; (403) 282-0095 (fax).

The first author thanks the Sloan Foundation Consortium on Competitiveness and the AEC Future Fund program for financial support. Insightful comments from David Mowery, Trond Petersen, J. C. Herb Emery, Joanne Oxley, Brian Silverman and Jeff Church and two anonymous referees are gratefully acknowledged.

Product Variety and Firm Survival in the Microcomputer Software Industry

Abstract

This paper provides an analysis of product variety and scope economies in the microcomputer software industry by using detailed firm-level and product-level information on firms' bundling of functionalites over application categories and computing platforms. We find that the management of product variety through the way different application categories are integrated in products and the platforms on which these products are offered can be as important as the significance of scope economies at the more aggregated firm level. Specifically, we find that there is little evidence of firm benefits from economies of scope in production, but there is substantial evidence that products benefit from economies of scope in consumption. In addition, we find that firms with products that encapsulate more application categories perform better, and those with products that cover more computing platforms perform worse. Finally, changes in product variety through new product introductions improve firm performance, but extensions to existing products hinder the performance of the firm and the product. We conclude that research in scope economies can benefit from a more detailed model of the evolution of product variety that includes data and analysis at the firm level and at the product level.

I. Introduction

The software industry has witnessed a wide assortment of product strategies over its short history, ranging from multiple stand-alone niche applications to integrated suites of applications. Scope economies play a critical role in the success of different product variety and product integration strategies. In this paper, we study firms' firm-level and product-level strategies in the production and consumption of microcomputer software in the early 1980s. We explore the impact of scope economies, product integration or focus, and product variety on firm and product performance. We examine these impacts in the context of application categories (e.g., spreadsheets, word processing, databases, etc.) and hardware/operating systems platforms (e.g., MS DOS, Apple II, CP/M).

Firms often have broad flexibility in their product-level implementation of a product variety strategy, and this is widely recognized in the software industry. Such flexibility includes bundling of different application categories into products, the offering of products over different computing platforms, introduction of new products into old or new categories and for old and new platforms, and the extension of existing products into new categories or for new platforms. When firms have such flexibility in their product variety strategy, detailed knowledge of product-level implementation and its outcomes is as essential as knowledge of firm-level outcomes to evaluate the impact of scope economies and product integration or focus on performance.

We use detailed information on microcomputer software product offerings in the early 1980s in order to investigate the management of product variety in the context of scope economies and integration in software. We uncover that the management of product variety through the way different application categories are integrated in products and the platforms on which these products are offered can be as important as the significance of scope economies at the more aggregated firm level. Specifically, we find that the effects of scope economies in production differed from effects of scope economies in consumption: there is little evidence of firm benefits from economies of scope in production, but there is substantial evidence that products benefit from economies of scope in consumption. In addition, we find that firms with products that encapsulate more application categories perform better, and those with products that cover more computing platforms perform worse. Finally, changes in a firm's product variety through new product introductions improve firm performance, but extensions to existing products hinder the performance of the firm and the product.

The remainder of this section provides a short history of development in the microcomputer software industry in the 1980s. The next section reviews the literature and arguments underlying different relationships between product variety and scope economies. The section that follows provides the definitions and hypotheses used in the study, and the subsequent section details the empirical approach, data and methodology. Our fifth section describes our results, and the final section discusses the implication of our findings.

Development of Microcomputer Software in the 1980s

During the early stages of the microcomputer software industry, product developers were convinced of the benefits of scope economies. But the challenges of managing product variety in order to appropriate these benefits were significant as illustrated by the range of approaches taken during the early life of this industry. Product design occurs on two levels. First, it requires that firms determine how to bundle capabilities such as addition, multiplication, text manipulation, presentation of graphs and charts, search and replace commands, etc., in the development of software applications. Second, product design must determine how to make these capabilities available. One option is as a stand-alone product in a single application category. Examples of the most common stand-alone application categories circa 1980 were: spreadsheet (e.g., Visicalc), database (e.g., dBASEII), word processors (e.g., WordPerfect) and programming languages (e.g., MS-BASIC). Another option is as product suites that include several application categories.

In the early 1980s significant development effort was required to extend capabilities and diversify into new application categories. Once the development of new capabilities was complete, they might be incorporated into the firm's existing products or used to develop a stand-alone product in the new category. For example, when a firm offering only a word processor decided to develop database-related capabilities (e.g., sorting and searching), it may have incorporated these functions into its word processor, or offered a stand-alone database product. Scope economies in the production of software follows from the ability to spread fixed development costs across several product markets. (Nobeoka and Cusumano (1997) evaluate the product development implications.)

Early in the first decade of competition, software developers anticipated significant economies from combining different application categories into integrated products. By 1983, microcomputer software developer Lotus had introduced its 1-2-3 package, combining capabilities in spreadsheet, database, programming and graphics categories in a single package. Prior to 1-2-3's introduction, these capabilities were commonly provided in stand-alone products such as those noted earlier. Many developers sought to replicate Lotus' market dominance by extending the features of already established products. Ashton-Tate, for example, combined its successful stand-alone database product dBASE II with text and graphical capabilities to produce Framework. Adding capabilities from more categories is an example of increasing product-level scope coincident with increasing firm-level 'scope' at Ashton-Tate. In an effort to extend its market dominance, Lotus introduced Symphony, which supplemented the 1-2-3 function set with text processing and telecommunications capabilities. But the complexity of the product met market resistance, suggesting that there are limitations to the extension of product breadth.

These limitations to more extensive integration were anticipated by software developers who believed that customers would pay a premium for broadly functional, interoperable applications in a series of stand alone products designed to work effectively in concert. These products could be sold either individually or, as eventually occurred, in a product bundle or 'suite.' Hence, interoperability has become an alternative to integration. A decade later, product bundles such as Lotus' SmartSuite, Corel Office, or Microsoft Office play an important role in competition. This strategy of interoperability was also attempted early in the industry; for example, in Microsoft's first applications. In the early 1980s, Microsoft designed its suite of separate spreadsheet (Multiplan), word processor (Multi-Word) and business graphics (Multi-Chart) products as the Multi-Tool set. Products shared command terminology and menu structures, and Microsoft planned a macro-language programming tool that would improve interoperability. However, this approach stalled and the term 'multi' was retained only for Microsoft's spreadsheet, Multiplan.

II. Product Variety and Scope Economies

Scope Economies in Production

In an overview of market structure in multi-product industries, Bailey and Friedlaender (1982) emphasize firm operating characteristics in determining industry competitiveness. Their model shows that firm-level economies of scope are necessary for multi-product industries. In a contestable markets model, multi-product firms better survive the threat of potential single-product entrants. These scope economies arise as a result of excess capacity in the face of indivisibilities when diverse product lines better utilize sunk investments. This is consistent with Montgomery and Wernerfelt (1988), *" ... diversification is prompted by excess capacity in rent-yielding factors that are subject to market failure*."

Product family sharing of key components and assets is one method of obtaining scope economies. This takes the form of a product platform where the designs, components and other assets are shared by a set of products (Meyer and Utterback, 1993; Robertson and Ulrich, 1998; Krishnan and Gupta, 2001). Scope economies occur because platforms make it cheaper to tailor products to segments, and diversification results from building and extending platform capabilities. That is, platforms reduce incremental costs of product variety yielding scope economies. But as Krishnan and Gupta (2001) found analytically, use of platforms may result in the overdesign of low end products. Not surprisingly, there is prior research suggesting that diversification correlates with poorer performance; firms in the Montgomery and Wernerfelt (1988) model that elect to diversify most widely also expect the lowest average rents.

Transaction costs reasoning asserts that there must be some reason to own, rather than rent, the excess capacity that leads to scope economies. Teece (1982) argues that multi-product firms arise when a variety of products demand similar production capabilities but firms are unable to independently secure access to requisite productive assets. Teece believes that a necessary precondition for multi-product firms is not technological indivisibilities but rather incomplete contracting for use of these assets. This rationale follows from standard transaction costs arguments that if slack production capacity could be contracted without risk of opportunism, the firm would rent, rather than own, the productive asset. If there are scope economies in the production of software, then this reasoning must apply to product development.

The application of transaction cost reasoning to the economic organization of software firms establishes a rationale for scope economies in the development of software. Weak appropriability in the design and implementation of software algorithms suggests that product innovations will be owned rather than licensed (see Teece (1986) for the argument). And, while the software community clearly shared some innovations amongst developers (e.g., Merges, 1992), other innovations were proprietary and aggressively protected.[1] If a software producer has a significant cost advantage in programming both a spreadsheet and a word processor because routines in one can be inexpensively added to the other, the firm should organize to take advantage of what is learned in the first product to reduce the cost of the second. Because this knowledge is likely to be 'sticky' and difficult to transfer, there is an advantage to development within the firm. Thus, economies in the way software is produced yields scope advantages in the

[1] For example, Wallace and Erickson (1993) describe the unique way that Lotus 1-2-3 uses DOS interrupts, something discovered only after significant reverse engineering by Microsoft operating system developers.

product development process. And, these economies should provide a cost advantage to firms that offer a greater product variety over more markets, resulting in improved product survival and profitability.

There are several reasons to believe that such economies were important in the early software industry. First, some extremely successful firms provided significant product variety. Microsoft, for example, offered operating system, programming, word processing, spreadsheet and game software all by 1981. Further, the Multi-Tool 'suite' (Microsoft Word, Multiplan, and Chart) employed a common interface so that users could more easily access functional variety. Although some firms retained a single product/single application strategy, most eventually turned to greater variety whereby most software firms offered multiple products in several application categories.

Many studies of scope treat production technology differences homogeneously. For example, scope economies between any two SICs depends on the "distance" between the 4-digit SIC, ignoring inherent similarities in production technologies. But firms may employ a variety of technologies with varying degrees of transferability to other segment classifications. To incorporate heterogeneous technologies, we proxy differences in production technology as choice of computing platform. Firms with expertise in one platform (e.g., IBMPC) will not necessarily have the same expertise in another (e.g., Apple Macintosh). The effect of production technology choice on firm performance can be modeled with this record of the platforms on which the firm chose to offer products. Firms able to produce software for a number of platforms are diversified against the risk that any individual platform becomes obsolete; offering products on more platforms gives the firm a lower risk of exit. Similar reasoning applies to individual product lines.

However, maintaining products across a wider array of computing platforms consumes resources that might be otherwise available for new product development. The argument that platform variety depletes firm resources would not have been convincing to industry participants circa 1980 (see Sherman (1984)). There are several reasons to think, however, that this strain on resources was important. As Boehm (1981) shows, software development for an operating system in which the firm has no experience can introduce significant uncertainty into the process, degrading development performance. The complexities in tailoring a product for a specific platform mean that diversity for any particular product is likely to be costly to maintain. The cross-level effect depends on the firm's ability to deploy technical resources in one product across its product portfolio.

The benefits of scope economies should not be assumed to continue indefinitely. At both the firm and product levels, greater product variety leads to greater complexity in managing each product. Because firms in this industry have limited resources, particularly small entrepreneurial firms, product-level implementation decisions balance greater variety with increasingly complex products. This potential for heterogeneity makes detailed knowledge of the firm strategy essential (see Merino and Rodriguez (1998)).

Scope Economies in Consumption

In addition to scope economies in production, there may be consumption economies of scope in this market, where customers prefer to purchase a variety of products from the same vendor. Although complementary products can be manufactured independently, goods with strong technological interdependencies are commonly produced by a single firm.[2] Most theoretical models do not make the distinction between scope economies related to production and those that derive from scope effects in consumption, primarily because in economic theory, cost minimization and profit maximization are dual. In the software industry, though, this distinction is important. Consumers of software applications appear to have a preference for purchasing

---

[2] There are exceptions. When design specifications are well-defined and the efficient administration of contracts can be accomplished cheaply, concerns over transaction governance costs can be minimized. Argyres (1996) describes the use of information technology in the manufacture of the stealth bomber by Lockheed, Boeing and McDonald Douglas. The firms collaborate on the wing technology by writing detailed descriptions of each component, and requiring conformance with a software CAD model for each change.

product variety from a single vendor. This effect results from the way that the firm coordinates product design in two facets of software development: a standardized command menu and functional interoperability.

A standardized menu facilitates the use of a wide variety of software by providing users with a unified command structure. With appropriate coordination of the menu, early entrants realized demand-side (consumption) economies of scope by making a series of products that lower adoption costs of learning the firm's other software. For example, the menu system for Microsoft products contained common commands for file manipulation and printing. After adopting Microsoft Multiplan, a user could more easily adopt Microsoft Chart, since the command structure was similar. This advantage to incumbents is clearly seen by the magnitude of the opportunity cost a user incurs in learning to use a software application, a cost usually far greater than the purchase price of the software. If this common command structure is difficult to provide across a suite of products, variety is better embodied in integrated products; integration should provide an advantage for the same level of scope. If, on the other hand, it is better to coordinate the menu structure, firms that offer a suite of products, but coordinate interface menus across those products, should have an advantage; firm scope over product integration.

We believe that scope economies in software follow from characteristics inherent in the way software products are designed, created and used. The drive for functional integration in the early 1980s under products like Lotus Symphony or Ashton-Tate's Framework suggests that developers believed in both broad category scope and category integration. Integrated products offered greater benefits to consumers because ease of interoperability was valued more than depth in any individual function.

Consumption side economies are closely linked to the consumer's willingness-to-pay for interoperability; appropriating these benefits requires coordination. Hold-up in incomplete contracting to provide functional interoperability (e.g., output from a database into a word processor) may lead to long-term contracting hazards as the products progress through upgrade cycles. This also will result in consumer preference to purchase product variety from an integrated producer. If the interoperability interface is standardized, specialization economies favor products from separate producers, as seen in the traditional stereo components industry.[3] This alternative strategy was employed by manufacturers of non-integrated products, and required the developer to provide interoperability between disparate products, perhaps across the boundary of the firm. Functional interoperability allows specialized applications to share common data files or output. For example, a retailer may store customer records in a database, produce charts of patron buying behavior, maintain mailing lists, and integrate these results into a word processing document. But with the rapid technological advances in microcomputer hardware, attempts at industry standard interfaces came and went.

Although this level of integration was eventually facilitated by the operating systems level, in the nascent software industry interoperability was provided by each developer, and rarely between companies. For example, spreadsheet producers provided that files could be saved in alternative formats readable by other applications: the exchange file structure for Visicalc was known as Data Interchange Format (DIF). Microsoft's Multiplan offered a Symbolic Link format (SYLK) for file exchange between other spreadsheets as well as to import data into Microsoft Chart. Walden (1986) reports that the interest in inter-product data sharing became so important that Lotus placed proprietary information describing the structure of 1-2-3 files into the public domain. Although the OLE (Object Linking and Embedding) level of integration was possible in MS-DOS computing, it was not feasible for a large majority of consumers because of the level of sophistication it required of users.[4]

---

[3] The literature is extensive, and specific works include Church and Gandal (1992), Economides and Salop (1992), and Farrell and Saloner (1992).

[4] In today's computing environment where the operating system industry structure is highly concentrated, OLE objects `drag-and-drop' from one application to another in a seamless way.

The tradeoff between scope economies in production and consumption is highlighted in Sorenson (2002) where managers must choose between efficient production of a small number of products and maintaining a range of products tailored to consumer preferences. And the latter strategy faces diminishing returns: broad product lines are increasingly costly and product variety is less valuable as the total number of products on the market increases.

Scope in Industry Evolution

In an industry evolving as rapidly as microcomputer software, some account of product and firm change is needed. Scope advantages, and the benefits of product variety more generally, occur as an industry evolves, and the firms and products that must evolve with it. Traditional studies of industry dynamics examine change from a relatively aggregated level, typically using snapshots of diversification: Bailey and Friedlaender (1982), Evans (1987), Shapiro and Khemani (1987), Dunne, Roberts, and Samuelson (1988), Audretsch (1991), Mata and Portugal (1994), Lieberman (1989), Audretsch (1994), Dunne and Hughes (1994), Baldwin and Gorecki (1994), an exception is Bruderl, Preisendorfer, and Ziegler (1992). It is increasingly evident, though, that studies should account for the process as well as the outcome of diversification strategies (see Barnett and Carroll, 1995). Firms that fail at attempts to diversify take greater risk, and should be treated differently than firms that make no effort to diversify.

Many diversification studies are cross-sectional and do not give appropriate attention to organizational change histories or market evolution.[5] Because the microcomputer hardware market was rapidly evolving, firm structures and capabilities were required to evolve with technological innovation. In most empirical work, diversification is both a state and a process, and must be considered in the context of market structure. In a rapidly developing market, firms may remain undiversified because they are successful in their own markets or unsuccessful in

[5] Recent exceptions are Kesner and Sharma (1996) and Bergh and Holbein (1997).

entering others. A longitudinal study records the attempted changes in the diversification path in order to discriminate between these alternatives. The degree of technological and market uncertainty in the software industry also make it important to model attempts at change and the resulting effect on firm and product survival.

There is ample research to suggest that organizational change imposes an exit risk on the firm, and this has implications at both the firm and product level. At the firm level, an increase in product variety depletes firm resources and may blur the firm's competitive profile. At a product level, increasing product functionality exposes the product to the uncertainty of competing in a new market. At least in the period of change, the product, and to some extent, the firm, face a greater risk of exit.

Difficulties in Studying Scope Economies

It is puzzling that product diversification studies often fail to support scope economies, but there are several explanations for why this might be the case even if scope economies are important. First, any economic rationale for advantages to scope differs across industries, and aggregation across SICs might blur these effects. Teece's transaction cost argument for multi-product firms, for example, should be justified in the practice of software development rather than naively asserted for all industries.   Second, it may be that the studies of scope at the aggregate business unit or corporate level are not accounting for product-level implementation strategies. In forming hypotheses about scope economies in software, it is important to consider the likely sources of these benefits in order to determine the implications for firm and product strategy. This step is especially critical in light of the array of implementation strategies available to software developers, and calls for a research design that accounts for this heterogeneity. Third, in a rapidly changing industry, transient effects may overwhelm static efficiencies in scope, so that an effective research design must accommodate industry evolution.

In many studies of diversification and scope economies there is concern that managers are not acting in the interest of shareholders. Any research on product variety must be concerned with principal/agent issues in risk diversification, because optimal levels of diversification differ between shareholders and managers. As Amihud and Lev (1981) demonstrate, managers prefer to diversify personal risk at the expense of returns to shareholders.[6] The benefit to sale of two products is reduced variability because sale of two products provides a smoother revenue stream than either individually. This is commonly argued to be an advantage for capital allocation in diversified firms. Managers might be tempted to over-diversify to smooth revenues at the expense of profits. But in an entrepreneurial industry like microcomputer software, there are several reasons to believe that strategic decisions were made primarily by owner-managers, with the result that the firms in our study are less likely to over-diversify under a principal/agent rationale. Tarter (1996) reports revenues and number of employees in *Soft*Letter 100*, the largest microcomputer software firms measured by revenues. These firms averaged only 90 employees over 1985-1986, the last year of our study. The bottom half of the list averaged only 20 employees. However, in the thousands of firms in our study revenues were on average far less than the top 100, and we conclude that firms in our study had even fewer employees, and thus were closely monitored by owners. Further, the information asymmetries in high technology products virtually guarantees that owners will be highly invested in product-line decisions. For this reason, there is little concern that managers will inefficiently diversify beyond profit-maximizing levels merely to smooth the revenue stream and inefficiently prolong firm and product survival.

III. Definitions and Hypotheses: Scope, Integration and Focus

Definitions

[6] Although some dispute this interpretation (see Lane et. al, 1998).

The alternative ways products incorporate capabilities from different application categories illustrate the complexities of the management of product variety in the software industry. That is, firms not only decide what capabilities to develop, but also whether to integrate capabilities of several application categories into a single product or maintain stand alone products for individual application categories. This is made even more complex by the decision of whether to develop products across computing platforms, develop integrated products for some platforms and stand alone products for others, or concentrate on a subset of platforms.

We define a *niche* as a specific application *category* on a specific computing *platform*. A *product* may cover a number of niches by including the functionality of several application categories, and possibly multiple computing platforms.[7] Thus, the terms “category”, “platform”, “niche” and “product” have separate and specific meanings in our research context. In this context, we use the terms “category scope” and “platform scope” to refer to the breadth of application categories or platforms in which the product or firm competes. We use the term “category integration” to refer to the range of application categories a particular product incorporates. Finally, we use the term “platform focus” to refer to the range of computing platforms on which a particular product is offered. Figure 1 shows how these terms are used.

*** Insert Figure 1About Here ***

At the firm level Firms 1, 2 and 3 all have the same category scope and platform scope as they each offer software in three application categories and for three computing platforms, but with different category integration and platform focus. Firm 1 offers a product that incorporates database, graphics and word processing, and has high category integration. In contrast, Firm 2 offers separate products for each of these categories and has low category integration. Similarly, Firm 1 has a broad platform focus and Firm 2 has a narrow platform focus. Firm 3’s products

[7] “Computing platform” has a distinct meaning different from “product platform” discussed in Section III.

have medium category integration and an intermediate platform focus. Firm 4 has both narrow category and platform scope, and a necessarily low category integration and narrow platform focus.

At the product level Firm 1's Product 1 has a broad category scope and platform scope as the product incorporates the three application categories and is offered on the three platforms. In contrast, Firm 2 and 4's products have narrow category scope and platform scope. Firm 3's products have intermediate category scope and platform scope.

Hypotheses

Our hypotheses concentrate on scope economies, category integration, platform focus, and product variety. As we discussed in Section II, economies of scope in production is supported by standard economics in a multi-product firm whereby it is possible to spread fixed costs of development across multiple products. It is also supported by transaction cost theory which argues that because of incomplete contracts governing knowledge assets that can be used for more than one software product, firms organize to own that asset and produce multiple products. In addition, there is the more recent literature on product platforms, whereby designs, components, and other assets are shared by a variety of products.

H1: There are scope economies in the production of microcomputer software.

Pure scope economies in production should result in a positive impact of broad category scope and platform scope that a firm offers in its product portfolio. Broad category scope and platform scope at the firm level should contribute both to firm performance and to product performance. Because scope economies in production occur at the firm level, there is not a necessary connection between individual product survival, and broad category scope and platform scope at the product level. As we further discussed in Section II, the principal/agent issues in risk diversification - which would tend to increase the number of products a firm offers in absence of scope economies in production – are less critical in the microcomputer software industry because the firms are either owner-managed, or are sufficiently small for owners to monitor employee product offering decisions closely.

On the demand-side there are reasons to believe that there are also scope economies in consumption. During the early years of the microcomputer software industry, interoperability and standards were more likely to apply to products from the same firm. Consumers put a high premium on these features because they simplified computer work by being able to move readily between applications, and they significantly reduced consumers' costs of learning and training.

H2: There are scope economies in the consumption of microcomputer software.

Scope economies in the consumption of microcomputer software are more likely to be product-based rather than firm-based because of superior interoperability. These scope economies should result in broad category scope and platform scope at the product-level having a positive impact on product survival. In other words, if there are consumption scope economies in the software industry, firms that offer functional variety in application categories (e.g., both a word-processor and a spreadsheet) in a single product should achieve better performance than rivals that offer single application category products. If scope economies in consumption occur between products, then broad category scope and platform scope at the firm level may also increase product survival, and should increase firm survival.

A naive approach tests the effect of product variety on firm survival without consideration for the way this variety is bundled. To the extent that scope economies in consumption are based on compatibility across categories and platforms, these economies should also result in a positive impact of high category integration and of broad platform focus on product survival. Because firms offered functional variety in application categories either at the product level (as integrated products) or at the firm level (through a common user interface), our approach must accommodate both levels. If a product-level approach to scope were more effective, products that integrate application categories would outperform less varietied rival products and contribute to firm survival. If the cost of integrating functional variety across several software products (as opposed to an integrated products) is prohibitive during this time-frame, then an increased number of products should not enhance firm survival.

Interoperability and standards between different application categories can be implemented in two ways. One is through different products in different sets of application categories offered by a single firm, where the fact that the products are produced by the same firm allows for a common interface and data exchange. The other is through products that cover larger numbers of application categories so that interoperability and a common interface is incorporated as part of the integrated product design. The former argues for greater product variety through numbers of products, and the latter would result in fewer products but with higher levels of integration.

H3: Integration over application categories yields higher firm performance.

The advantages of integration over multiple products in delivering product variety should result in higher category integration being positively associated with firm survival and product survival, while controlling for category scope. That is, holding category scope constant, fewer products but integrated over application categories should enhance firm performance. H3 is stated so that the alternative to H3 is also testable – implementing product variety through multiple products rather than integration enhances firm performance.

As we discussed earlier in our overview of scope economies in production, there is evidence that maintaining products across computing platforms can be resource intensive due to the fundamental differences in operating systems. This is consistent with the higher costs of deriving product variants for multiple product platforms, rather than deriving product variants from a single platform.

H4: Focus on a smaller number of computing platforms yields higher firm performance.

The advantage of focus through offering individual products over a smaller number of platforms should result in narrower platform focus having a positive impact on firm survival and product survival, holding platform scope constant. Thus, a greater proportion of products that are offered over multiple computing platforms should lower firm performance.

Scope and industry evolution suggest that increasing product variety and new product introductions are important for firms to maintain their product diversification advantages. It also allows firm to dynamically adjust strategies such as integration versus multiple products, and follow industry evolution in application categories and computing platforms. In addition, accounting for industry evolution is an important control when examining effects such as those in our first four hypotheses.

H5: Changes in product variety increase firm performance.

Adjustments in product variety, scope, integration and focus strategies are adjustments in firm-level strategies. As measured by entry into new products, application categories, computing platforms, etc., these entries should have a positive impact on firm survival. To the degree that these new entries do not compete with existing products, they should also have a positive impact on product survival.

IV. Empirical Approach, Data and Methodology

Empirical approach

We study the effects of scope economies, integration, focus, and product variety at both the product level and at the firm level. Use of multiple levels of analysis is important in understanding the trade-offs at both the firm and product levels, and the effect of firm-level decisions on individual product performance. Product and firm survival are used as our two measures of performance. Survival is a particularly relevant measure of performance in high technology industries, having been used in studies of the computer workstation industry (Sorenson, 2000) among others. In a time of rapidly expanding markets, participation in one year of the industry can be seen as an option to continue in later, more profitable, years. Exit from the industry identifies firms for which the option to continue was worth less than the opportunity cost of remaining in the industry.

Data

Two data sources are used to provide data for our analysis. Detailed information on individual software product data was assembled from *The Software Catalog: Microcomputers* published by Elsevier Publishing (1980).[8] The catalog lists thousands of software packages available by hundreds of firms in the U.S. market. The data covers the years from 1981 through 1986 at 9 catalog publication dates. This time period is a critical period in the microcomputer software industry since it spans the pre-IBM PC era through the introduction of the Apple Macintosh. During this period the number of products available grew from 3,700 to 13,000 and the number of companies from 500 to 6,000. In *The Software Catalog: Microcomputers* each product is classified as incorporating one or more specific application categories and as offered over one or more computing platforms. We combined some of the classifications from *The Software Catalog: Microcomputers* for our analysis, for example, text processing and spell checking software are combined with word processing. Table 1 lists the application categories and platforms used in the classification. Use of various definitions of category and platform do not qualitatively alter the results. For example, using category classifications of word processors, database, graphics, operating systems, spreadsheets, telecommunications, programming, games, personal and accounting did not change the qualitative results of our analyses.

*** Insert Table 1 About Here ***

To provide for a measure of market size for microcomputer software, product market data on dollar microcomputer sales collected by International Data Corporation (IDC) is used as a proxy of the potential market for software products.

---

[8] The 1981/82 data was published by Imprint Software Ltd.

Dependent Variables

Performance is measured by product survival and firm survival. Observations for a product (or firm) begin when the product (or firm) first appears in *The Software Catalog: Microcomputers*. These observations continue until the product (or firm) no longer appears in *The Software Catalog: Microcomputers*. The two variables that measure survival are *Product Exit* and *Firm Exit*. These are coded as zeros during the observations when the product (or firm) appears, and are coded "1" when the product (or firm) exits.

There are at least three reasons why survival is a good measure (Sorenson, 2002). First, survival is a necessary condition for positive profits. As such, survival is really a threshold measure. Second, survival (or exit) can be observed for all products and all firms. Third, exit directly measures poor performance.

Operationalizations of Scope

Category Count and *Platform Count* are used to operationalize category scope and platform scope, at both the product and firm levels. At the product level these variables are the number of application categories (e.g., word processor, spreadsheet) in which a given product is classified, and the number of computing platforms (e.g., Apple II, Apple Macintosh, IBM-PC) that a product is classified under. At the firm level these counts are the number of application categories in which the firm has a product classified, and the number of computing platforms under which the firm has at least one product classified.

A *Niche Count* is used to operationalize an additional dimension of scope at the product level. This is the number of niches, defined as the intersection of category and platform, in which the product is classified. A *Product Count* is used to operationalize our final dimension of scope at the firm level. This is the number of products that the firm has on the market.

Operationalization of Integration and Focus

The operationalization of integration and focus uses two measures of entropy: category and platform entropy. Entropy measures have long been advocated in studies of diversification, but rarely applied to product-level variety in the way necessary here.[9] An exception is Alexander (1997) who measured the relationship between industry structure and product variety in the music industry, measuring variety by an entropy index.

Category Entropy is used to operationalize category integration and *Platform Entropy* is used to operationalize platform focus, both at the firm level. We use the standard method of calculating entropy used elsewhere (e.g., Alexander, 1997) where at the product level the calculation is $- \sum [p_i \ln(p_i)]$ where p_i is probability that the product is classified in a particular category (platform) and the sum is over the categories (platforms). The average over the firm's products is the *Category* (*Platform*) *Entropy*. Category entropy measures the tendency of the firm to offer integrated functionality such that firms that tend to offer products integrated over multiple categories have greater category entropy. Platform entropy measures the extent to which firms tend to offer a product for several platforms such that firms that tend to offer each product across several platforms have greater platform entropy.

Using platforms to illustrate the distinction between a count and the entropy measure, suppose that two firms offer products for the same number of platforms, but one firm offers each product for only one platform, while the other offers each product for several platforms. The firm offering each product for only one platform has a low entropy measure and the other has a high entropy measure.

Operationalizations of Changes in Scope

To operationalize the dynamics of scope changes at the product level three counts are used. *New Category* (or *Platform*) *Entry* is a count of the number of new categories (or platforms) the

---

[9] Jacquemin and Berry (1979), Palepu (1985) and Hoskisson, Hitt, Johnson, and Moesel (1993) discuss the use of entropy measures in firm-level diversification.

product was made available for during the "spell" – that is, during successive publications of *The Software Catalog: Microcomputers*. In addition, *New Category and Platform Entry* is a count of the new category/platform combinations the product was made available for during the spell.

New Category Entry and New Platform Entry for products are also aggregated to the firm level. At the firm level we have three additional variables. *New Product Entry* is a count of new products introduced. *New Product in New Category* and *New Product in New Platform* are counts of categories and platforms in which new products are introduced, summed over the new products.

Control Variables

Control variables are used to account for industry and competitive effects. To account for differences in overall software market potential and opportunity across periods, *Microcomputer Sales* (in $Millions) is used as a measure of market size.

To account for the competitive environment a set of different counts of products classified in the same niche, category, and platform are used as measures of competition. The intuition for these measures is that customers make a series of nested binary comparisons, where more firms or products in the relevant market implies, all other things equal, greater competition for each product. At the product level *Products in Niche* is a count of products in the same niche, for example, all word processor applications for the IBM PC, or all spreadsheets for the Apple II. If the product is classified in multiple niches, then Products in Niche is a count over the different niches. *Products in Category* and *Products in Platform* is a count of the number of products in the same categories or platforms, respectively.

At the firm level *Firms in Category* and *Firms in Platform* is a count of the number of firms in the same categories or platforms, respectively. In both the product-level and firm-level measures, no adjustment is made for a competing product or firm that is classified in more than one of the same categories or platforms. Although this may cause our measures to be overstated relative to measures such as the 'number of competing products or firms', classification of a competing product in more than one category or platform means competition across those categories or platforms.

Table 2 provides descriptive statistics on the product-level and firm-level variables. To determine if multicollinearity is a problem we examined the correlation matrix's condition number to determine if it was 'ill-conditioned' and it is not. In addition, the effect of higher correlations would be to understate the significance of our coefficients, so our results are above any variance inflation factor.

*** Insert Table 2 About Here ***

Methods

We employ event history methods to model the survival of software products. Petersen (1991) recommends a logit model of spell-splitting with time-varying covariates to estimate a non-proportional hazards model, and Yamaguchi (1991) recommends the non-proportional hazards model to the continuous time models frequently employed in these kinds of studies. Following these recommendations, we use a discrete-time logit model for our duration equation. Details of the mathematical form are in Yamaguchi (1991:18-19). Survival is a censored observation (=0) and exit is an event (=1); thus for interpreting the results, negative coefficients correlate with a decreased exit probability, positive coefficients correlate with increased exit. Whenever the discussion identifies exit risk, it should be understood to mean an increase in the probability of exit with an increase in the covariate.

The statistical analysis below provides survival models of firms, products and the effects of firm-level covariates on product survival. We use two methods for model selection when there is concern over multicollinearity. When the number of covariates is relatively small (e.g. less than 15), we employ the best subsets approach advocated in Sen and Srivastava (1990). For a larger number of covariates, the computational time was significantly higher, and the 'selection method' (iteratively applied forward and backward selection) was employed to determine covariate inclusion. After a set of coefficients was selected, these were included across models to provide a consistent set in the discussion.

V. Results

Age and Survival

In our firm-level analysis Model 1 in Table 3 shows firm survival with age of the firm as the only covariate. In the absence of information about firm heterogeneity, older firms are shown more likely to survive. Later models, however, show that in the presence of firm scope, integration and focus, changes in scope, market size, and competitive environment, age of the firm increases the probability of exit. Many studies of organizational mortality indicate that older firms have a survival advantage, but it has also long been suspected that unobserved heterogeneity might account for this (see Bruderl, Preisendorfer, and Ziegler (1992), Mayer, Hamerle, and Blossfeld (1989), Mayer and Tuma (1990)). In the presence of covariates modeling heterogeneity, the true effect in our data is a liability of obsolescence, a result consistent with expectations about high technology industries. Model 2 of the table includes more detailed information on firm heterogeneity. Model 3 is a statistically significant improvement over model 2 and, since the results are consistent, our discussion of the firm level analysis draws on Model 3.

*** Insert Table 3 About Here ***

In our product-level analysis Model 1 in Table 4 shows statistical models of product survival, and Model 2 shows these effects in the presence of firm-level covariates. Inclusion of firm-level covariates significantly improves model fit, even after accounting for the increase in model degrees of freedom; firm-level effects are important for modeling product survival. As with firms, products show a liability of obsolescence.

*** Insert Table 4 About Here ***

The age of the firm improves an individual product's likelihood of survival. Firm age may signal the likelihood of longer-term product upgrades and maintenance. Customers may be more willing to sink the investment in learning new software with some assurance that the investment will be recouped by future upgrades or availability on new platforms. The effect of this covariate is opposite from the firm level of analysis, where firms are subject to a liability of obsolescence.

Scope and Survival

At the firm level of analysis, Model 3 in Table 4, the coefficient of product count shows a greater probability of exit for firms with more products, and the coefficient of category count shows a greater probability of exit for firms that have products in a greater number of application categories. Both of these results are statistically significant and indicate diseconomies of scope in the production of microcomputer software at the level of the firm. The coefficient for platform count is negative, correlating with a decreased probability of exit for firms that have products under a greater number of computing platforms. Although this is suggestive of economies of scope in computing platforms, this result is less statistically significant than that for products and categories. In addition, the coefficient is only a fifth the size of the coefficient for category count, yielding a mean effect that is less than a third of the mean effect of categories.[10]

Examining the firm-level variables at the product level of analysis, Model 2 in Table 4, the coefficients of product count, category count, and platform count are all positive and statistically significant. This indicates that the exit of any individual product is more likely when the firm has multiple products, any individual product is less likely to survive when the firm has products in more categories and when the firm has products under more platforms. Thus, in

---

[10] Using the means for the counts from Table 2, multiplied by the coefficients from Table 3.

terms of product survival, there are diseconomies of scope at the firm level in products, categories, and platforms. With reference to our first hypothesis (H1), we can conclude that there are not scope economies across products or application categories in the production of microcomputer software. There may be positive but small scope economies in production across computing platforms at the firm level, although individual product survival does not benefit from these economies. This difficulty in managing product market scope fits with the popular sense that industry participants were far more interested in computing technology than consumer markets. Understanding the emerging markets for software applications, measured by the scope of product markets, appears to have been a challenge to firm survival.

At the product level of analysis (Model 2, Table 4), the product-level variables for category count and platform count have coefficients that are negative and significant. This indicates that products which cover a greater number of application categories and products which operate under a greater number of computing platforms each have a greater probability of survival. This provides evidence in support of our second hypothesis (H2): at least at the product level, there are scope economies in the consumption of microcomputer software whereby products that have functional variety that covers more application categories and products that can be used on more computing platforms are more likely to survive. The effect of a product covering a greater number of niches (categories by platforms) is negative – a higher niche count reduces the probability of product survival. This reflects limits of scope economies of consumption. However, it is worth noting that the mean effect of category count is almost twice that of niche count, so that the combined effects of these measures of product scope increase the probability of product survival.[11]

Integration, Focus and Survival

---

[11] Ibid, except coefficients are from Table 4.

As detailed earlier, we use measures of entropy to operationalize category integration and platform focus, and these are firm-level measures: larger category entropy indicates that the firm tends to offer integrated products, and a larger platform entropy corresponds to the firm offering each product on a larger number of computing platforms.

In the analysis of firm survival (Model 3, Table 3) the coefficient for category entropy is negative and significant. This indicates that greater category entropy increases the probability of firm survival. Therefore, firms that offer integrated products have an increased chance of survival. In the analysis of product survival (Model 2, Table 4) the coefficient of firm-level category entropy is also negative and significant. So, in addition to firm survival, greater category entropy increases the probability of product survival. Thus, when the firm tends to offer integrated products, each individual product has an improved survival probability.

Further supports the argument for consumer preference for integrated products over inter-operable suites comes from the product-level category count in Model 2, Table 4. The negative coefficient indicates that products with greater category scope have a greater probability of survival, at least in the time-frame of our study. Therefore, our results support our third hypothesis (H3) – namely that integration over application categories yields higher firm performance.

In contrast to the result on category integration, the positive and significant coefficients for platform entropy in the firm-level analysis in Table 3 (Model 3) and in the product-level analysis in Table 4 (Model 2) shows that greater platform entropy correlates with increased exit both for firms and for products. Therefore, for firms which offer products over a greater number of computing platforms, the individual product is less likely to survive, and the firm is less likely to survive.

The results on platform entropy are consistent with results of firm platform scope at the product level (Model 2, Table 4) – products of firms that cover a greater number of platforms are less likely to survive. These results support our fourth hypotheses (H4) that a firm's focus on a smaller number of computing platforms yields higher performance.

Change in Product Variety and Survival

We begin with the firm survival analysis of Model 3 in Table 3. At the firm level there are two types of adjustments to product variety: new product introductions and extensions of existing products to new application categories and/or to new computing platforms. The coefficient of new product entry, reflecting the number of new product introductions, is negative and significant. This indicates that the greater the number of new products introduced, the greater the probability of firm survival.

However, the coefficients of new category entry and new platform entry are both positive and significant. These variables reflect the firm's extension of one or more of its existing products to new categories and to new platforms (new to the firm, not new to the industry), and these product extensions are correlated with firm exit. At the level of analysis of the firm our fifth hypothesis (H5) has equivocal support: changes in product variety in the form of new products increases firm performance, while product extensions decrease firm performance.

In the analysis of product survival in Model 2 of Table 4, the product-level variables of new category entry and new platform entry also have positive coefficients, indicating that extending the product to a new application category or to a new computing platform is correlated with product exit. The coefficient for new category and platform entry is significant and negative, but when combined with the positive (exit) effects of both new category entry and new platform entry, the overall effect is a lower probability of product survival. Thus, at the product level, extensions into new categories and new platforms are correlated with product exit.

The firm-level variables in the product survival analysis indicate that individual product exit is more likely in a period when the firm introduces a new product, and this is accelerated when the newly introduced product is a diversification into a new category. The firm-level effect of entering a previously unserved platform improves product survival probabilities. The coefficient signs suggest that new products cannibalize old ones, so that unless the firm appeals to new set of customer needs on a different platform, new product introductions lead to product exit. More dramatic diversification by producing a new product in a new category or new platform correlates with improved survival – such a new product entry would not be a competing product.

Controls

In the firm survival analysis (Model 3, Table 3) the negative and significant signs of the coefficients of the environmental variables show that the larger the number of firms in categories and the greater number of firms in platforms, the less likely is firm exit. Moreover, the greater the size of the market (measured by microcomputer sales), the less likely is firm exit. This suggests that the market size was growing faster than the suppliers of software over the duration of the study, so much so that additional competitors was an indication of a good category or good platform to have products in.

In the product survival analysis, (Model 2, Table 4) more products in category and more products in platform correlates with increased product survival, more products in niche correlates with decreased product survival. At this product level, products in niche more precisely measures inter-product competition. The positive sign of the products in niche coefficient reflects the greater exit probability when there are more products in the same functional categories on the same platforms. This is analogous to local and non-local density measures (see for example Baum and Korn (1996), Baum (1995), Baum and Singh (1994)).

VI. Discussion and Implications

This study contributes to our understanding of scope, integration, focus and product variety. The regressions tell a consistent story about scope economies: although firm-level scope measures do not provide evidence of scope economies in production, at the product level they provide support for scope economies in consumption. In addition, the dispersion (entropy) measures demonstrate that category integration and platform focus is important: high application category integration increases survival of the firm and the firm's products, and similarly for a narrow computing platform focus. Finally, adjustments in product variety though new product introductions and product extensions have distinctly different implications for firm and product survival: the introduction of new products increases the probability of firm survival, and extensions of existing products to new categories and new platforms reduces the likelihood of both firm and product survival.

The contrast between the effects of our operationalizations of scope at the firm level – increased probability of firm exit with greater product scope and category scope, and at the product level – increased probability of product survival with greater category scope and platform scope – are stark. We believe this confirms the distinction between our first two hypotheses. Scope economies in production, our first hypothesis, should affects firm survival by spreading fixed production costs across products, and should not affect product survival as the allocation of those cost savings to individual products is arbitrary. Scope economies in consumption, our second hypothesis, are demand-side effects where the impact is on individual product survival.

Implications of our study beyond the software industry certainly depend on the nature of the product variety decisions. Scope economies have always been important to software, but the integration decision critically determines the extent to which users access this scope. As Porter notes, the ability of customers to assemble their own bundles is an important determinant of the extent of scope economies realized. The strength of integrated products over the time-frame of the study was in two key attributes: command interface and interoperability. No matter how well developed the interface to different products, during the era of MSDOS the extent of product interoperability required significant user expertise. It is ironic that the industry has turned to product suites in a time when command interfaces and interoperability are increasingly standardized. Conformance to standardized command structures appears to be far stronger under a graphical user interface than under character-based operating systems. Functional interoperability has also been dramatically standardized through Object Linking and Embedding (OLE). Although OLE was a long-standing goal of operating system developers, its implementation in a broadly accessible, intuitively appealing way was long and labored.

Challenges in supporting a larger number of computing platforms draw attention to the question of the firm's choice of platform focus. Greater diversity in computing platforms appears to benefit firm survival. However, the standardization of the installed base of hardware to a few computing platforms – platform focus - foreshadows the eventual importance of facility with a particular computing environment. Generalist approaches to platform development, where software was available for any and all platforms, were lauded early in the industry. Eventually, however, these were not competitive as bandwagon effects resulted in an industry with fewer platforms. However, it remains unclear whether founding as a specialist in a single platform would have been the best approach. It may be that a generalist strategy provides greater flexibility against the eventual narrowing on a few platforms. Even if firms widely anticipated platform dominance by IBM, it isn't obvious what the best strategy would have been.

Our analysis across firm and product levels provides an additional perspective on firm evolution. Our results are the first to combine detailed product knowledge with firm-level characteristics, and demonstrate the necessity of multiple levels of analysis in the study of scope economies. This cross-level analysis supports a 'flagship' model of competition also referenced in Nobeoka and Cusumano (1997). Firms such as Lotus (1-2-3), Ashton-Tate (dBASE II), Borland (Turbo Pascal), and Novell (Netware) all had a central successful product that was complemented by other product lines, consistent with the studies of product platforms as the basis of product variety discussed in Section II.

Our results also demonstrate that care should be taken to evaluate the importance of firm-level initiatives in the context of individual product strategies. The age of the firm, perhaps as a signal of longer-term maintenance and upgrades, enhances survival of individual products. This provides some mechanism for the finding in many previous studies that older firms have a lower hazard rate. Perhaps surprisingly, the number of other products offered by the firm decreases the probability of survival of any individual product. This greater probability of product exit may be due to divided interests, with a broad portfolio resulting in/from a lack of commitment to any particular product initiative.

This research also shows that studies of scope economies should control for process of product diversification. As we found, adjustments in product variety can have positive or negative effects on firm survival depending on whether they are new product introductions or product extensions: introductions of a new product correlates with a decrease in exit, while entering a new category or platform increases exit. It is likely that the new product effect is due to the nature of innovation in product development that is so important in this industry versus the extension of a product that is already in the process of exit.

Although many characteristics of the firm play a significant role in survival of any individual product, the influence of firm change on individual product survival is dramatic; firm expansion is not frictionless. In our analyses, firm expansion in products and application categories produces increased risk of product exit, but this is compensated by access to greater opportunities.

Scope economies and focus strategies have been evolving throughout the history of the software industry. Management of product variety has to account for these changes. Clearly, the relationship between scope and integration/focus has changed over the past 20 years, particularly as the operating system facilitates greater product market and platform integration. In the emergent microcomputer software industry, integrated products had an advantage, but the results might be very different in today's computing environment. The evolution in product bundling and its implications for economies of scope remains an important issue.

This research has attempted to separate scope economies in consumption from scope economies in production by giving attention to the industry context. Even greater attention to a model of the customer should provide the rationale for both where scope economies should be expected and why. Research that ignores the production and consumption aspects of industry context will miss this important subtlety. The value we found of separating firm-level from product-level effects suggests that a much more detailed model of scope economies is warranted, one that pays attention to these different sources of scope economies.

The levels of analysis in this study provide an unprecedented look at product and firm survival. Integration and focus strategies affecting the way the firm appropriates the benefit of scope economies may matter more than aggregate benefits of scope more generally. Managing product variety and complexity in the firm may be all the more important, so that the existence of scope economies is less critical than the ability to manage product and firm evolution in an advantageous way. Damanpour (1996) suggests that greater complexity facilitates innovation in larger organizations. The insight provided in our software study suggests that product-level measures of complexity are equally important. And in this context, determination of the characteristics of production and consumption that leads to these benefits is critical.

VII. References

Alexander, P.J., `Product variety and market structure,' *Journal of Economic Behavior& Organization*, **32**, 2 (February), 1997, pp. 207-214.

Amihud, Y. and B. Lev, `Risk reduction as a managerial motive for conglomerate mergers', *Bell Journal of Economics,* **12,** 1981, pp. 605-617.

Argyres, N. "Information technology and organizational boundaries: Evidence from the stealth bomber", Academy of Management Annual Meetings; Cincinnati, OH. (1996)

Audretsch, D.B. `New-firm survival and the technological regime', *Review of Economics and Statistics,* **73,** 1991, pp. 441-450.

Bailey, E.E. and A.F. Friedlaender, `The theory of oligopoly with multi-product firms: Market structure and multiproduct industries', *Journal of Economic Literature,* **20,** 1982, pp. 1024-1048.

Baldwin, J.R. and P.K. Gorecki, `Concentration and mobility statistics in Canada's manufacturing sector', *Journal of Industrial Economics,* **42,** 1994, pp. 93-103.

Barnett, W.P. and G.R. Carroll. `Organizational change.' In J. Hagan and K.S. Cook, (eds.) *Annual Review of Sociology.* Annual Reviews, Palo Alto, CA, 1995, pp. 217-236.

Baum, J.A.C. `The Changing Basis of Competition in Organizational Populations: The Manhattan Hotel Industry, 1898-1990', *Social Forces,* **74,** 1995, pp. 177-204.

Baum, J.A.C. and H.J. Korn, `Competitive Dynamics of Interfirm Rivalry', *Academy of Management Journal,* **39,** 1996, pp. 255-291.

Baum, J.A.C. and J. Singh, `Organizational Niches and the Dynamics of Organizational Mortality', *American Journal of Sociology,* **100,** 1994, pp. 346-380.

Bergh, D.D. and G.F. Holbein, `Assessment and redirection of longitudinal analysis: Demonstration with a study of the diversification and divestiture relationship', *Strategic Management Journal,* **18,** 1997, pp. 557-571.

Boehm, B.W. *Software engineering economics,* Prentice-Hall, Englewood Cliffs, NJ, 1981.

Bruderl, J., P. Preisendorfer, and R. Ziegler, `Survival chances of newly founded organizations', *American Sociological Review,* **57,** 1992, pp. 227-242.

Church, J. and N. Gandal, `Integration, complementary products, and variety', *Journal of Economics & Management Strategy,* **1,** 1992, pp. 651-675.

Damanpour, F. `Organizational complexity and innovation: developing and testing multiple contingency models', *Management Science,* **42,** 1996, pp. 693-716.

Dunne, P. and A. Hughes, `Age, size, growth and survival: UK companies in the 1980s', *Journal of Industrial Economics,* **42,** 1994, pp. 115-140.

Dunne, T., M.J. Roberts, and L. Samuelson, `Patterns of firm entry and exit in U.S. manufacturing industries', *Rand Journal of Economics,* **19,** 1988, pp. 495-515.

Economides, N. and S.C. Salop, `Competition and integration among complements', *Journal of Industrial Economics,* **40,** 1992, pp. 105-123.

Elsevier Publishing, *The Software Catalog: Microcomputers,* Elsevier, New York, 1980.

Evans, D.S. `The relationship between firm growth, size, and age: Estimates for 100 manufacturing industries', *Journal of Industrial Economics,* **35,** 1987, pp. 567-581.

Farrell, J. and G. Saloner, `Converters, compatibility, and the control of interfaces', *Journal of Industrial Economics,* **40,** 1992, pp. 9-35.

Hoskisson, R.E., M.A. Hitt, R.A. Johnson, and D.D. Moesel, `Construct validity of an objective (entropy) categorical measure of diversification strategy', *Strategic Management Journal,* **14,** 1993, pp. 215-235.

Jacquemin, A. and C. Berry, `Entropy measure of diversification and corporate growth', *Journal of Industrial Economics,* **27,** 1979, pp. 359-369.

Kesner, I.F. and A. Sharma, `Diversifying entry: Some ex ante explanations for postentry survival and growth', *Academy of Management Journal,* **39,** 1996, pp. 635-677.

Krishnan, V., and S. Gupta, “Appropriateness and impact of platform-based product development,” *Management Science*, **41**, 1 (January), 2001, pp. 52-68.

Lane, P. J., Cannella Jr, A. A., and M.H. Lubatkin, `Agency problems as antecedents to unrelated mergers and diversification: Amihud and Lev reconsidered', *Strategic Management Journal*, **19**, 6 (June) 1998, pp. 555-578.

Lieberman, M.B. `The learning curve, technology barriers to entry, and competitive survival in the chemical processing industries', *Strategic Management Journal,* **10,** 1989, pp. 431-447.

Mata, J. and P. Portugal, `Life duration of new firms', *Journal of Industrial Economics,* **42,** 1994, pp. 227-245.

Mayer, K.U., A. Hamerle, and H. Blossfeld, *Event history analysis: statistical theory and application in the social sciences,* L. Erlbaum Associates, Hillsdale, N.J. 1989.

Mayer, K.U. and N.B. Tuma, *Event history analysis in life course research,* University of Wisconsin Press, Madison, WI, 1990.

Merges, R.P. `A comparative look at property rights and the software industry', *WP - Boston University,* 1992, pp. 1-52.

Merino, F. and D.R. Rodriguez, `A consistent analysis of diversification decisions with non-observable firm effects', *Strategic Management Journal,* **18,** 1997, pp. 733-743.

Meyer, M.H. and J.M. Utterback, `The product family and the dynamics of core capability', *Sloan Management Review*, **34**, 3 (Spring) 1993, pp. 29-49.

Montgomery, C.A. and B. Wernerfelt, `Diversification, Ricardian rents, and Tobin's Q', *Rand Journal of Economics,* **19,** 1988, pp. 623-632.

Nobeoka, K. and M. Cusumano, `Multiproject strategy and sales growth: The benefits of rapid design transfer in new product development', *Strategic Management Journal,* **18,** 1997, 169-186.

Palepu, K. `Diversification strategy, profit, performance, and the entropy measure', *Strategic Management Journal,* **6,** 1985, pp. 239-255.

Petersen, T. `Time-aggregation bias in continuous time hazard rate models.' In P.M. Marsden, (ed.) *Sociological methodology.* Oxford: Basil Blackwell, 1991.

Robertson, D. and K. Ulrich, `Planning for product platforms', *Sloan Management Review*, **39**, 4 (Summer), 1998, pp. 19-31.

Sen, A. and M. Srivastava, *Regression analysis: Theory, methods, and applications,* Springer-Verlag, New York, NY, 1990.

Shapiro, D. and R.S. Khemani, `The determinants of entry and exit reconsidered', International *Journal of Industrial Organization*, **5**, 1987, pp. 15-26.

Sherman, C.E. *Up and running: Adventures of software entrepreneurs,* Ashton-Tate Inc. Culver-City, CA, 1984.

Sorenson, O., "Letting the market work for you: An evolutionary perspective on product strategy," *Strategic Management Journal*, 32, 5 (May), 2000, 577-592.

Tarter, J. *Soft*Letter 100,* Soft*Letter, Watertown, MA, 1996.

Teece, D.J. `Towards an economic theory of the multiproduct firm', *Journal of Economic Behavior and Organization,* **3,** 1982, pp. 39-63.

Walden, J. *File formats for popular PC software: A programmer's reference,* John Wiley & Sons, Inc. NY, NY, 1986.

Yamaguchi, K. *Event History analysis,* Sage, Newbury Park, CA, 1991.

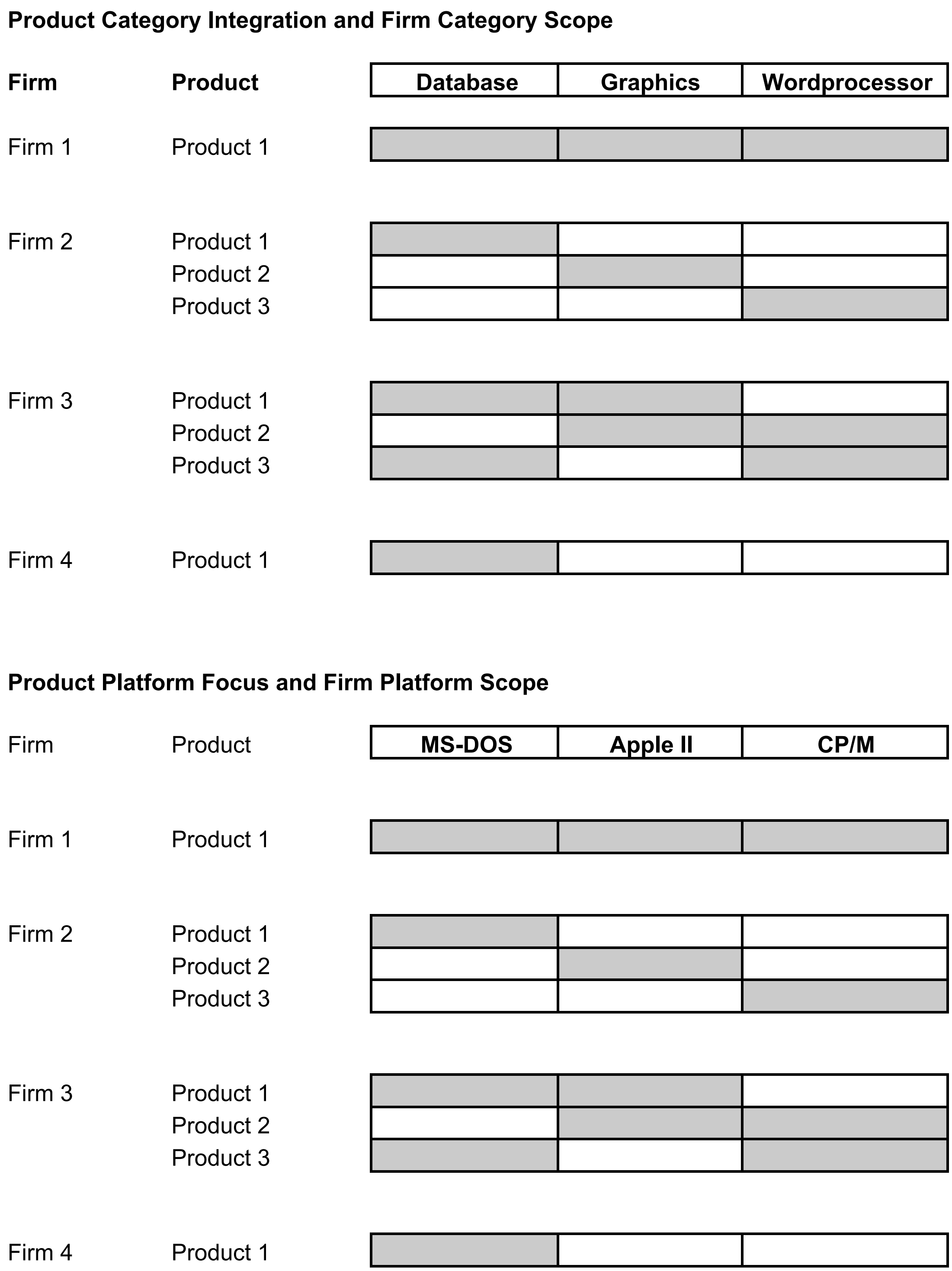
Product Category Integration and Firm Category Scope
Firm
Product
Database
Graphics
Wordprocessor
Firm 1
Product 1
Firm 2
Product 1
Product 2
Product 3
Firm 3
Product 1
Product 2
Product 3
Firm 4
Product 1
Product Platform Focus and Firm Platform Scope
Firm
Product
MS-DOS
Apple II
CP/M
Firm 1
Product 1
Firm 2
Product 1
Product 2
Product 3
Firm 3
Product 1
Product 2
Product 3
Firm 4
Product 1

Figure 1

Table 1: Software classification definitions

Category	Platform
Educational	Apple II
Accounting	IBM - Personal Computer
Vertically Integrated industry software	Digital Research CP/M family
Entertainment	Radio Shack/Tandy
Programming	Victor/Commodore/PET
Operating System software	Other Intel family of microprocessors
Business Administration	Atari
Miscellaneous personal software	BASIC language programs
Engineering Tools	Hewlett-Packard family of micros
Word Processors	Digital Equipment Corporation
Communications software emulators	Northstar family of microcomputers
Database	Other IBM (non IBMPC)
Graphics	Altos
Misc. Scientific	Motorola
Information Retrieval	Texas Instruments
Other	

Table 2: Descriptive Statistics

Variables	Mean	Std Dev	Min	Max
Product Level: N = 117242				
Product Exit	0.012	0.220	0	1
Left Censored	0.270	0.896	0	1
Product Age - 1 Year=100	148.207	126.285	25	550
Category Count	1.168	0.410	1	10
Platform Count	1.837	1.536	1	13
Niche Count	2.133	2.168	1	41
New Category Entry	0.042	0.213	0	3
New Platform Entry	0.139	0.688	0	10
New Category and Platform Entry	0.012	0.109	0	1
Products in Category	6147	5549	18	38018
Products in Platform	1229	1627	1	12926
Products in Niche	2271	1844	19	12700
Microcomputer Sales ($M)	7.738	4.615	2.390	9.704
Firm Level: N = 22210				
Firm Exit	0.027	0.163	0	1
Left Censored	0.182	0.386	0	1
Firm Age - 1 year = 100	166.282	131.138	25	550
Product Count	3.987	5.860	1	50
Category Count	3.032	1.655	1	16
Platform Count	3.071	1.513	1	12
Category Entropy	0.099	0.213	0	1.6094
Platform Entropy	0.437	0.561	0	2.4849
New Product Entry	0.265	1.309	0	20
New Category Entry	0.137	0.502	0	11
New Platform Entry	0.186	0.659	0	8
New Product in New Category	0.090	0.583	0	14
New Product in New Platform	0.113	0.888	0	20
Firms in Categories	4047	1477	273	12085
Firms in Platforms	5138	2101	48	11725

Table 3: Firm survival

Variables	Model 1 coeff	Model 1 stderr	Model 1 sig	Model 2 coeff	Model 2 stderr	Model 2 sig	Model 3 coeff	Model 3 stderr	Model 3 sig
Constant	-3.663	0.038	**	-2.270	0.100	**	-2.412	0.104	**
Left Censored				-0.443	0.072	**	-0.426	0.073	**
Firm Age	-0.0008	0.0002	**	0.003	0.0003	**	0.003	0.0003	**
Scope									
Product Count				0.001	0.006		0.019	0.006	**
Category Count				0.143	0.029	**	0.307	0.034	**
Platform Count				-0.048	0.023	*	-0.060	0.024	*
Integration & Focus									
Category Entropy				-0.390	0.115	**	-0.488	0.115	**
Platform Entropy				0.756	0.072	**	0.789	0.073	**
Product Variety									
New Product Entry				-0.070	0.023	**	-0.058	0.023	*
New Category Entry				0.158	0.058	**	0.136	0.058	*
New Platform Entry				0.031	0.020		0.048	0.021	*
Controls									
Firms in Categories				-0.0002	0.00002	**	-0.0003	0.00002	**
Firms in Platforms				-0.0002	0.00002	**	-0.0002	0.00002	**
Microcomputer Sales($M)				-0.1693	0.02530	**	-0.1815	0.02580	**
Administration							-0.037	0.030	
Education							-0.242	0.057	**
Entertainment							-0.070	0.027	*
Operating Systems							-0.260	0.071	**
Vertical Industry Applications							-0.289	0.079	**
Altos							0.087	0.018	**
Digital Research-CP/M							-0.097	0.014	**
IBM PC							-0.035	0.025	
North Star							-0.079	0.023	**
Tandy/Radio Shack							-0.053	0.012	**

	Model 1	Model 2	Model 3
LR X^2	18	1,143	1,308
df	1	13	23
Concordant	31%	67%	68%
Discordant	43%	30%	28%
Tied	26%	4%	3%

Statistical significance	**	*1%*
	*	*5%*

Table 4: Product survival with firm-level covariates

Product-Level Variables	Model 1 coeff	stderr	sig	Model 2 coeff	stderr	sig
Constant	-1.9630	0.054	**	-2.5701	0.058	**
Left Censored	-0.2927	0.025	**	-0.2995	0.029	**
Product Age	0.2100	0.0094	**	0.4070	0.015	**
Scope						
Category Count	-0.2668	0.030	**	-0.2218	0.032	**
Platform Count	0.0428	0.016		-0.0548	0.018	**
Niche Count	0.0574	0.009	**	0.0618	0.010	**
Product Variety						
New Category Entry	0.5043	0.037	**	0.4739	0.03760	**
New Platform Entry	-0.0094	0.010		0.0522	0.01290	**
New Category and Platform Entry	-0.2611	0.077	**	-0.2323	0.07760	**
Controls						
Products in Category	-0.0001	0.000	**	-0.0001	0.000	**
Products in Platforms	-0.0001	0.000	**	-0.0001	0.000	**
Products in Niche	0.0001	0.000	**	0.0001	0.000	**
Microcomputer Sales ($M)	-0.1222	0.006	**	-0.1150	0.006	**
Firm-Level Variables						
Firm Age				-0.0021	0.0002	**
Scope						
Product Count				0.0136	0.001	**
Category Count				0.0710	0.006	**
Platform Count				0.0686	0.005	**
Integration & Focus						
Category Entropy				-0.4810	0.082	**
Platform Entropy				0.3067	0.032	**
Product Variety						
New Product Entry				0.0332	0.003	**
New Category Entry				0.0634	0.015	**
New Platform Entry				-0.0887	0.009	**
New Product in New Category				-0.0339	0.005	**
New Product in New Platform				-0.0082	0.002	**

	Model 1	Model 2
LR X^2	3,135	5,229
df	12	23
Concordant	62%	66%
Discordant	35%	31%
Tied	3%	2%

Statistical significance ** *1%*

□ * *5%*