Adoptions and Orphans in the Early Microcomputer Market*

forthcoming Journal of Industrial Economics

Neil Gandal, Department of Public Policy, Tel Aviv University, 69978 Tel Aviv, Israel, e-mail gandal@econ.tau.ac.il,
Shane Greenstein, Department of Management & Strategy, Kellogg Graduate School of Management, Northwestern University, Evanston, IL 60208, e-mail s-greenstein1@nwu.edu, and
David Salant, LECG, 2000 Powell St. Suite 600, Emeryville, CA 94608, e-mail David_Salant@lecg.com

September 1, 1998

Abstract

In this paper we examine the development of the microcomputer market in the early 1980s. CP/M, a widely-adopted operating system, was orphaned by the user and development communities. A new operating system, DOS, and a new hardware platform, the IBM PC, became the predominant industry standard. We examine the statistical relationship between data that reflects hardware and software sales for the competing platforms. We conclude that the economic processes underlying the development of DOS differed from those underlying CP/M and that many of these differences related to the role of software development.

JEL Classification Numbers: L86, O33.

*We are especially grateful to Severin Borenstein and two anonymous referees for suggestions that significantly improved the paper. We also thank Richard Arnott, Tim Bresnahan, Francesca Cornelli, Gregory Duncan, Raphael Rob, Abi Schwartz, Yishai Yafeh and seminar participants at the 1994 Winter Econometric meetings, the University of Pennsylvania, and INSEAD for helpful comments. We received outstanding research assistance from Subhendu Roy and Susan McMaster. Greenstein would like to acknowledge partial funding from NSF IRI-92-09321.

I Introduction

In this paper we examine the development of the microcomputer market in the early 1980s. CP/M, a widely-adopted operating system, was orphaned by the user and development communities. A new operating system, DOS, and a new hardware platform, the IBM PC, became the predominant industry standard. While this market episode has generated many overlapping analyses and theories, it has largely escaped systematic statistical analysis. This is not so surprising. Many empirical implications of standardization models depend on unobservable features of competitive behavior, vendor costs or consumer preferences. In addition, these episodes tend to happen in new or incipient markets. It is very rare for any such market episode to lend itself to direct measurement of the factors underlying positive feedback. In this paper we bring some data to bear on the analysis of the PC market standardization, a case in which positive feedback played an important role.

Our research strategy is to examine the statistical relationship between data that proxies for sales of operating systems and software equipment associated with competing platforms. In our analysis, we measure the positive feedback (in the statistical sense of a vector auto regression) between different components in competing systems. We find econometric evidence that in the case of the CP/M, there was two-way feedback of a similar magnitude between the operating system and applications software. We also find evidence that the success of DOS depended on the availability of compatible software, and vice versa. However, we find that for DOS the feedback from operating systems to software development was significantly stronger and more sustained than vice versa. We conclude that the economic processes underlying the development of DOS differed from those underlying CP/M and that many of these differences related to the role of software development.

This case is of interest because it is an important example of orphaning arising out of competition between competing standards. Orphaning occurs when late adopters choose a technology incompatible with the technology adopted by early users, and suppliers of supporting services (complementary products) cease to provide their products for the old technology. More generally, orphaning is of concern to vendors and users in electronics markets where technical standards and product designs are fluid.

Orphaning occurs, in part, because of the heterogeneity in adopter cohorts. Early and late buyers may make different choices when there are significant changes in the availability of complementary services over time. Often, high value, technically skilled users are the first to adopt new technologies. In the primitive stage of the market, when only these technically skilled consumers (techies) can use a new product, one standard may be better, in the sense that it is part of a better system without complementary products. But it may be the case that another product dominates as a mature product, that is, with a well developed base of software and other complementary peripheral products. If there is uncertainty about whether these products will ever develop a network of complementary products, then it is possible that techies will adopt the product that works best without complementary products. If a much larger and better network of complementary products develops for another product, it may be adopted by non-technical users who tend to adopt later. Such a scenario leads to orphaning of the technically-oriented product.

The key observation is that early and late buyers may make different choices when there are significant changes in the availability of complementary services over time. The switch in platforms is facilitated by a significant increase in the availability of complementary products for the new platform. These observations lead to two key ideas for empirical purposes. If there was indeed a significant change in the availability of complementary software for the two platforms, this should be observed in the data. In other words, the historical record should show a different pattern of feedback between software and operating systems for the early and late platforms. In addition, the historical record should show that the orphaning of an early technology correlates with differences in the behavior of early and later consumers and potentially producers.

We take this analysis to the early microcomputer market, where CP/M was the de facto standard operating system and was subsequently replaced by the DOS operating system. Our paper employs advertising data on the microcomputer market in the early 1980s. We examine the role played by increases in the availability of software and peripherals for the competing platforms and vice versa. During the period we investigate, two new operating systems were deployed to take advantage of new 16 bit chips, CP/M - 86 and PC (or MS) DOS. CP/M-86 was an extension of the earlier 8 bit version of CP/M that had dominated the microcomputer market. PC/MS-DOS was developed for the IBM PCs. For a few years during the early 1980's both operating systems were widely distributed and competed directly against each other.

We find that for the case of CP/M, lagged software advertising significantly predicts advertising for microcomputers using the CP/M operating system, and that lagged advertising for microcomputers running the CP/M operating system significantly predicts software advertising: there is two way feedback between the operating system and software and the magnitudes are similar. We observe a different relationship in the case of DOS. While there is again two way feedback between the operating system and software for the case of DOS, the feedback from the operating system to software is much stronger and more persistent. This suggests that the eventual adoption of DOS as the industry standard was due to the significant number of complementary software packages and peripherals available for it; we believe that the significant amount of software available for DOS was due to the strong ties between IBM (the hardware producer) and the software development community. This highlights differences in consumers over time and the heterogenous roles firms may play in episodes of positive feedback.

Our paper adds to the empirical literature on platform competition and technology adoption in the computer industry. Gabel [1991] and Langlois and Robertson [1992] provide an extended economic history of the personal computer industry. The latter identify factors leading to open platforms in the long run, while the former is a detailed case study concerning the role of *de facto* standardization on the evolution of the microcomputer industry. In contrast to these two studies, our empirical work mixes the statistical with the descriptive. Our empirical analysis builds on Bresnahan and Greenstein [1997a, 1997b], who characterize platform competition in the first three decades of the computer industry. We also add to the few empirical studies, such as Gandal [1994], which provide empirical evidence that the value of a hardware system depends on the variety of complementary software.

II Platform competition in the microcomputer market

It is not our intention to recount all the detail behind the economic history of the personal computer industry.[1]Here we explain the basic factors that shaped platform competition between the CP/M and DOS operating systems and discuss several alternative explanations.

II(i) The sequence of events

Several vendors sold machines to customers between 1975 and 1980. The vast majority of the early users of these machines tended to be, and perhaps needed to be, computer literate. This group included many tinkerers and hobbyists. These are the techies. The benefits derived by the early users from their microcomputers were to a great extent a function of the user's ability to experiment and program. This contrasted with later users who tended to be less sophisticated and more interested in standard business applications.

CP/M was the dominant open operating system during this early period. No firm solely sponsored this platform in 1980, nor had any single firm taken responsibility for its development in the past. CP/M was a combination of technical standards, an operating system and shared application software from other hobbyists and many small companies. No single computer maker controlled the interface standards, operating system, or hardware architecture. Many firms made machines that ran the CP/M operating system. Software for this platform was (mostly) able to run on the computers from any of these firms. Customers could mix and match hardware, software, and peripherals.

The other preeminent operating system was the partially-open/partially-closed Apple II. This system was designed and produced by Apple, though Apple was willing to let users add some peripherals and software from other companies. Apple served a hobbyist market from the beginning and never really successfully moved into the business market, as did CP/M and DOS (See Gabel [1991], and Langlois [1992]). Apple's failure with the business market occurred in spite of the appearance of spreadsheet program first on an Apple (i.e., Visicalc) and attempts to upgrade the system with the Apple III. As with CP/M, Apple sold primarily through retail stores and mail-order houses.

There is a literature that argues that DOS succeeded, in part, because DOS was open and Apple was closed. In this view, openness refers both to features of the hardware architecture and to the relationship of the chief sponsoring firm, Apple and IBM, with other software developers.[2]

Any user could add hardware or software components to an IBM-compatible PC. This opened possibilities for expanding the capabilities of the systems. While the choice of an open or closed platform is an important dimension of competition, our interest is more with documenting and analyzing the factors that determined the success and failure of the two open platforms, CP/M and DOS, where the operating systems/software feedback patterns did not solely arise out of a vertically integrated firm structure.[3]

The introduction of the IBM PC occurred in late 1981. The IBM personal computer used a 16-bit chip from Intel and the DOS operating system. The chip represented an increase in performance over the 8-bit chips predominantly in use in all previous PCs, both in the Apple II and in those using the CP/M operating system. Users had a choice between versions of DOS supplied by IBM and Microsoft.[4]The two versions of DOS were very similar. Most early buyers of IBM PCs used the IBM version, which was a bit more compatible with the IBM proprietary BIOS (the basic input/output system embedded in the chip). The advantages of the IBM BIOS were slight, so over time, and as clones appeared, an increasing number of PCs were sold with MS DOS.

As Langlois and Robertson [1992] document, the change in platforms underwent two phases. One occurred beginning in the 1981-1982 period, when CP/M and DOS operating systems competed and the outcome was uncertain. For example, the July 1982 edition of Byte magazine devoted 26 pages to an analysis of the two 16 bit operating systems (the CP/M-86 and MS-DOS) competing for dominance.[5]During this phase, some non-techies entered the market. Many systems of both type were sold, despite the competition between (and uncertainty about) the two platforms.

By the 1984-1985 period, which was well into Langlois and Robertson's second phase, the IBM PC with the MS-DOS operating system had supplanted the CP/M machines. The IBM PC and DOS operating system could be found in most major US businesses by then. By 1984, Gabel [1991] notes that there were 11,000 different software programs available for the MS-DOS operating system. The CP/M operating system was effectively dead by 1986.[6]

Thus, a snap shot of the industry in 1985 hardly resembled a snap shot of the industry in 1980. The primary users were technically sophisticated in 1980. They were general purpose (business oriented) by 1985. The main applications were limited in 1980 and often were not user friendly. In 1985 applications were varied and many emphasized their ease-of-use for non-technical users. And most interesting for our purposes, the dominant technical standards embedded in the operating systems of the majority of PCs in 1980 differed from those embedded in the majority of PCS in 1985. Since most of the new application software was incompatible with the CP/M systems, a large fraction of the users of PCS in 1980 found themselves orphaned by 1985.

II(ii) Alternative interpretations

Many different explanations have been offered for the success of the IBM PC and DOS platform and the failure of CP/M. Many of these explanations overlap. It is not our intention to test among them, which would be a hopeless exercise since all of them contain elements of truth to them. Instead, we will show that our data are consistent with two of these explanations.

- *Availability of Complementary Software:* One reason offered for the early success of CP/M, and then its later orphaning, was availability of applications software. In this view, the success of DOS and the IBM PC in later years again depended on the availability of software and peripherals. Much of this software came from firms other than IBM and Microsoft. Once the applications software for MS-DOS materialized, there was general agreement that MS-DOS was a better and faster single-user single-tasking operating system for nontechnical users. As documented by Gabel, this was certainly established by 1985 and as early as 1984. In fact, it almost certainly was established sooner in some circles.[7]As the number of non-technical users in business began to grow in 1982-83 in response to the software available, their numbers swamped the number of techies.

  It is well known that most remaining software and peripheral developers for CP/M abandoned CP/M between 1983 and 1984 and moved to DOS-based applications. By 1985 virtually no more applications were being written for CP/M;

due to the availability of software, DOS had become a superior platform for both techies and non-techies.

- *Technical Superiority:* It was not obvious that DOS was technically superior to CP/M. Both CP/M-86 and DOS were 16 bit operating systems and the technical merits of the two operating systems were hotly debated in the technical community at the time (though this debate is often forgotten after the tide that accompanied DOS by the mid 1980s). For example, in the detailed comparison of the CP/M-86 and the MS-DOS operating systems that appeared in Byte magazine in 1982, Richard Lomas, a system manufacturer noted that both DOS and CP/M operating systems could be run on the early Intel 8086 chips, seemingly giving users the option of porting old CP/M applications to the new IBM hardware. Furthermore, Lomas noted that the CP/M software was also compatible with MP/M-86, a multi-user system. The upward compatibility of MS-DOS software to Xenix (the multi-user system specified by Microsoft) was less certain.

  The above arguments supporting CP/M's technical superiority would have had the greatest appeal with technical users. Indeed, many of the techies continued to adopt CP/M systems in the 1981-1982 period. Yet, most IBM hardware was delivered to users packaged with PC-DOS (IBM's version) or MS-DOS (Microsoft's version); a user in search of CP/M operating system would have had to seek out CP/M operating system with their new microcomputer. While this was not a barrier to most experienced techie users, it was one more difficulty for non-technical users to overcome. For a user starting from scratch or with little installed base of CP/M application software, the relative merits of using the CP/M operating system might have seemed less compelling as soon as a lot of DOS-based software became available.

- *The relationship between IBM and the Business Community:* Another explanation for the success of the IBM PC and DOS emphasizes IBM's unique position within the computer industry, especially its strong market and marketing position with large businesses and with the software development community. IBM had a tremendous existing customer base in traditional data processing shops throughout large corporations. Its existing marketing and support network initially viewed the PC as a complement to already established mainframe networks, where most users had experience with terminals. PCs could act as intelligent terminals, and with a bit of technical gerrymandering at first, and less so as IBM improved the system software, could transfer data from mainframes to small applications on the PC. When user-friendly spreadsheets, databases and wordprocessors appeared on DOS, these PCs were able to perform simple analytical and word-processing tasks while by-passing capacity constraints associated with the use of a central data base on a mainframe. The Techie-oriented systems that preceded the IBM PC were less able to address both sets of needs. Additionally, the installed base of customers perhaps also provided greater assurances to developers of software who IBM encouraged to develop software for DOS-based systems in the early 1980s.[8]

Not all of these views are testable nor are they mutually exclusive. Accordingly, our empirical goals are modest and our test is somewhat indirect. The hypothesis about software availability suggests that increasing the presence of CP/M or DOS software should make the relevant operating system more valuable, inducing further purchases. In that case, it complements the feedback from the adoption of operating systems to software, i.e., the purchase of the operating system induces future software purchases. In contrast, the hypothesis about technical superiority suggests that feedback between operating systems and software was not an essential part of the competitive process.

Thus, in this view there is likely to be little difference between the feedback of DOS and CP/M. Finally, the hypothesis about IBM's unique position suggests that the feedback between operating systems and software ought to differ between DOS and CP/M. DOS would have to have a stronger process of feedback in order to overcome the advantages accruing to the incumbent system, CP/M.

We show that data are consistent with the explanations that emphasize the importance of complementary products and with explanations that emphasize the relationship between IBM and the software development community. We first show that there was two way feedback between the operating system and software for both CP/M and DOS. We then show that the software development response to the adoption of DOS operating system was much more significant than the feedback from the operating system to software development for CP/M. This led to the significant amount of software available for the DOS platform, which enabled its rapid adoption.

III Empirical Evidence

III(i) The Data

We now briefly describe data used in this study. We collected quarterly data on the number of pages of advertisements in Byte magazine. We use microcomputer advertisements as proxies for operating system sales. The fact that the microcomputer ran a particular operating system was clearly stated in the advertisement. If such an advertisement indicated that the microcomputer ran several operating systems, each operating system was credited with an equal proportion of the advertisement. Software and peripherals were advertised as being compatible with DOS and or CP/M operating systems. If an advertisement for one of these products indicated compatibility with several platforms, each platform received an equal proportion of the advertisement.[9]Advertisers included producers of personal computers (such as Compaq and IBM), as well as software producers (such as SSI - later Word Perfect - and Lotus). Figures 1 and 2 show (respectively) the advertising data for microcomputers running each operating system and for software that was compatible with each operating system.[10]

Figures 1 and 2 about here

We chose Byte because, unlike other computer magazines, Byte is a general magazine that covered developments for all operating systems. In addition, despite growth in the overall market (and slightly in the size and circulation of the magazine), it largely did not alter its format over the time period we examine it, nor did it veer from providing coverage for the whole PC market. Thus, it provides us with a natural base-line for computing component market share.

We believe that the number of pages of software and peripheral advertising is a reasonable proxy for the relative number of complementary products available for a particular operating system.[11]Given that a computer system typically consists of a single microcomputer and a single operating system, we believe that advertisements for microcomputers running particular operating systems are a good proxy for the sales of the operating systems.[12]We will also show below that many of the basic features of these data correspond with other descriptions of the market's development, as represented in several descriptive histories of the time.

These data have strengths and weakness for our purposes. Its main strength is that it provides a quantitative and consistent indication of the growth, commercial success, and failure of all the categories of components associated with these different computing platforms. Because it is so difficult to construct consistent measurement of new or incipient markets, this may be the only measure that can do so. Its main weakness is that there is no generally accepted theory of advertising for high-technology markets, nor any systematic empirical literature on the topic.[13]So there is no commonly accepted way of relating advertising to the rate of sales, or the installed base. Thus, there is a maintained, but basically untestable, premise throughout this study that advertising closely tracks actual sales. While this is plausible, this inherent limitation suggests that we should take care below not to overstep our interpretation. We now describe our data in detail.

We track the CP/M market from April 1978 to October 1986, which is almost the entire lifetime of products associated with the platform. We track the DOS platform from July 1981, the date any product on the DOS platform was first advertised, to October 1986. We stop at this point primarily because the advertising associated with products using the CP/M platform is so scattered and rare as to no longer warrant much interest.[14]We collect quarterly observations, which results in 36 and 22 complete observations for CP/M and DOS respectively.

Table I presents some basic summary statistics and the figures display histories. As shown in figures 1 and 2, advertising for the DOS platform grows over the entire period, while advertising for the CP/M platform peaks around 1982-3. Total advertising grows over the whole period, reflecting the entry of many new consumers into this market. The growth and death of total advertising conforms closely to industry perceptions about the growth and death of all five platforms, which we take as evidence that total advertising tracks commercial activity.

It is useful to compare that the relative amounts of software and peripheral advertising with advertising for the microcomputers running the operating systems. The summary statistics in Table I show there was a much higher proportion of software and peripheral advertising relative to advertising for microcomputers running the operating systems. This most certainly reflects real economic behavior, that is, users continue to buy software for an operating system after the initial purchase.[15]

Table I about here

These data provide additional evidence that the level of these advertisements positively correlates with real economic activity. In particular, figures 1 and 2 are consistent with the preeminence of the CP/M and Apple platforms during the late 1970s and early 1980s. Additionally, figure 2 shows that as late as 1983, software vendors using CP/M invested almost as heavily in advertising as those using DOS; this is consistent with the evidence that many techies adopted CP/M systems in the 1981-2 transition period. Finally, the data show that advertisements for microcomputers running CP/M ceased to exist by the 1984-5 period and software advertisements had declined significantly. All these patterns are consistent with the historical record.

III(ii) Econometric Analysis

Given the limitations of our data, we are restricted to testing for predictability or causality in the narrow, technical sense formalized by Granger (1969) and Sims (1980). In this interpretation, a variable x causes y if lagged values of x are significant in explaining y in a regression in which lagged values of y are also explanatory variables. It is, of course, possible that causality can exist in both directions. This test can be performed using vector autoregessions (VARs). We are not estimating a structural model when performing these tests; nevertheless, we believe that this type of analysis is useful in assessing whether there are differences in the advertising patterns for DOS and CP/M.

Table II presents results from VAR regressions of microcomputers running operating systems on software and visa-versa. Because there is no natural specification for the different effects of software and peripherals, we examined three alternative non-nested specifications of the software/peripheral variables. In the first model, we used all three variables (hardware, software, and peripherals). In the second model, we used only hardware and software. In the third model, we added peripherals to software, so that the variables were hardware and "software + peripherals". All specifications fit relatively well. For brevity, in Table II we show estimates using the second model. The results were not qualitatively different for the other two models.[16]We conclude that the hardware/software specification provides a reasonable and parsimonious description of the market's change over time.

Table II about here

First we examine the question: what is the relationship between lagged values and contemporaneous values of microcomputers running operating systems and compatible software. In table (II), we report results from VARs with two lags.[17]

This table shows that in the case of CP/M, single lags of software significantly predict later hardware advertising, controlling for lagged hardware advertising. Similarly, a single lag of hardware advertising significantly predicts software advertising. The magnitudes of the feedback effects are similar in both directions and in both cases second lags are insignificant in both directions. We thus conclude that there is two way feedback between the operating system and software in this case and the magnitudes of the feedback are similar.

We observe a different relationship in the case of DOS. Table (II) shows that both first and second lags of hardware advertising are very large in magnitude and significant in predicting software advertising. In the case of the feedback from software to hardware advertising, the results are quite similar to CP/M. A single lag of software significantly predicts later hardware advertising, controlling for lagged hardware advertising, but the second lag is insignificant. Hence, although there is two-way feedback, the feedback is such that the effect of lagged hardware on software availability is much stronger and more persistent than vice versa.

In order to summarize the information in the VARs, we used the estimated VARs to calculate impulse response functions. The methodology is straightforward. Ignoring the error term, a VAR with two lags can be written as

$$Y_t = c + \alpha_1 Y_{t-1} + \alpha_2 Y_{t-2} + \beta_1 X_{t-1} + \beta_2 X_{t-2}. \quad (1)$$

Using the familiar lag operator, L, Equation (1) can be rewritten

$$A(L)Y_t = c + B(L)X_t, \quad (2)$$

where c is a constant and $A(L) = (1 - \alpha_1 L - \alpha_2 L^2)$ and $B(L) = (\beta_1 L + \beta_2 L^2)$. An impulse response function with the variable X as the impulse and Y as the response is defined by

$$Y_t = k + D(L)X_t = k + (\delta_0 + \delta_1 L + \delta_2 L^2 + ... + \delta_m L^m)X_t, \quad (3)$$

where k is a constant. Substituting (3) into (2),

$$A(L)k + A(L)D(L)x_t = c + B(L)x_t. \quad (4)$$

If the (α and β) parameters are known, then the distributed lag (impulse response) structure is found by setting $D(L) = B(L)/A(L)$ and solving for δ.[18]The results are that $\delta_0 = 0, \delta_1 = \beta_1, \delta_2 = \beta_2 + \alpha_1\beta_1$, and $\delta_3 = \alpha_1(\beta_2 + \alpha_1\beta_1) + \alpha_2\beta_1)$. The results with numerical values appear in table III.[19]

Table III shows that for both CP/M and DOS, the impulse responses functions calculated from the VARs when the operating system is the response (and software the impulse) die out after a single lag. Indeed, it is clear from Table III that there is no statistical difference between DOS and CP/M regarding the feedback from software to the operating system. On the other hand, the impulse response functions calculated from the VARs when software is the response (and the operating system is the impulse) are quite different for the two platforms.

Table III about here

The software response function for CP/M also dies out, although perhaps slightly slower than the response function for the operating system.

For the software impulse response function for DOS, we see a completely different pattern. The magnitude of the one lag effect is nearly twice the one lag effect for all the other impulse response functions. More importantly, the function does not die out. Instead it increases so that the second lag effect is much greater than the already large first lag effect. The second lag effect here is much larger than the second lag effect for the other impulse response functions. Finally, even the third lag effect is very large.[20]

The patterns associated with DOS software advertising contrasts with those associated with CP/M. Our interpretation is that the while the success of DOS depended largely on the availability of software, the software development response to increases in DOS operating systems dwarfs that of CP/M; this effect, which is likely due to IBM's relationship with software developers, enabled DOS to supplant CP/M. Hence, while the eventual success of DOS largely revolved around the entry of many DOS-based software vendors, our empirical results suggest that IBM's ability to leverage its large market share in traditional data processing to encourage rapid very development of complementary products for its PCs.

The development of the partially-open Apple platform was quite different from that of the two open systems we considered in the text. Table IV shows that there is no economically strong relationship between software and hardware for the Apple platform. This suggests that the nature of product development at Apple lead to much less predictable hardware/software interrelationships.

Table IV about here

In summary, increases in CP/M or DOS software tends to lead to increases in advertising for microcomputers running the particular operating system. From this we can infer that applications software makes operating systems more valuable. However, in addition, we see that the increase in advertising for microcomputers running a particular operating system also increases the amount of software available for the operating system, consistent with two-way positive feedback in this market. While we cannot reject the hypothesis about the technical superiority of CP/M or DOS, we find stronger evidence in favor of views emphasizing feedback. Finally, we find that the feedback from hardware to software differs significantly between DOS and CP/M. This suggests strongly that IBM's unique position within the industry gave it an advantage when organizing software developers around the new hardware system.

The data are consistent with the explanations that emphasize two way feedback between software and operating systems. They are also consistent with the importance of complementary products and with explanations that emphasize the relationship between IBM and the software development community. This led to the significant amount of software available for the DOS platform, which enabled its rapid adoption as the industry standard.

IV Conclusion

We examined the development of the microcomputer market in the early 1980s. CP/M, a widely-adopted operating system, was orphaned by the user and development communities. A new operating system, DOS, and a new hardware platform, the IBM PC, became the predominant industry standard. We examined the statistical relationship between advertising for microcomputers running a particular operating system and applications software compatible with the same operating system. We concluded that the economic processes underlying the development of DOS differed from those underlying CP/M and that many of these differences related to the role of software.

Late adopters of PCs chose systems that were incompatible with the technology adopted by early users. Suppliers of supporting services (software and peripherals) ceased to provide their products for the old technology. Orphaning occurred, in part, because of the heterogeneity in adopter cohorts. Early and late buyers made different choices, and these choices depended on significant changes in the availability of complementary services over time. Orphaning also occurred, in part, due to heterogeneity in developer cohorts. The system that eventually won was able to amass a greater feedback of software developers for its platform than any previous platform.

The PC industry is not unique in the electronics industry in its basic market structure. Many other markets also combine irreducible technical uncertainty, early technically sophisticated users and less technical later adopters, and a network of complementary suppliers. Many other industries also contain significant elements of platform competition. Thus, many emerging markets could display orphaning.

Choosing the wrong technology has costs for both suppliers and users; it is a danger that many different players guard against. Managers of firms that sponsor platforms spend considerable resources managing networks of suppliers. Small suppliers within a network spend considerable resources understanding their role within the network, trying to gain increasing influence over the direction of whole (e.g., see the recommendations in Ferguson and Morris [1993]). Some analysts argue that orphaning becomes less likely as the market matures and standard designs emerge (e.g.,Steffens [1994]), so firms begin to guard less against its occurrence and focus on other immediate problems.

Whatever one's view, this feature of market structure dominates many dramatic events of recent times and merit further analysis. We have taken the first steps towards bringing empirical content to the analysis of these events. It points towards further research into the role of heterogenous users and heterogeneous producers in models of standardization. We also look forward to further analysis of the structural determinants of positive feedback behavior using modern statistical tools.

References

[1] Bresnahan, T., and S. Greenstein, 1997a, "Technical Progress and Co-Invention in Computing and in the Use of Computers," *Brookings Papers on Economics Activity: Microeconomics*, Pp. 1-78.

[2] Bresnahan, T., and S. Greenstein, 1997b,"Technological Competition and the Structure of the Computing Industry," Center for Economic Policy Research, Stanford University, Publication No 315.

[3] Cringely, R. X., 1992, Accidental Empires, New York: Harper Collins.

[4] Ferguson, C. H. and C. R. Morris [1993], Computer Wars: How the West Can Win in a Post-IBM War, 1st ed., New York: Times Books: Random House.

[5] Friedman, A., and D. Cornford [1989], Computer Systems Development: History, Organization and Implementation, John Wiley and Sons, New York, NY.

[6] Gabel, L., 1991, Competitive Strategies for Product Standards, Mc-Graw Hill, UK.

[7] Gandal, N., 1994, "Hedonic Price Indexes for Spreadsheets and an Empirical Test for Network Externalities," *RAND Journal of Economics,* 25: 160-170.

[8] Granger, C., 1969, "Investigating Causal Relations by Econometric Models and Cross-Spectral Methods," *Econometrica,* 37: 424-438.

[9] Greene, W., 1993, "Econometric Analysis," Macmillan Publishing Company, New York.

[10] Klenow, P., 1994, "New Product Introductions," Mimeo, University of Chicago.

[11] Langlois, R., and P. Robertson, 1992,"Innovation in Modular system: Lessons from the Microcomputer and Stereo Component Industries," *Research Policy,* 21 (4), 297-313.

[12] Prusa, T., and J. Schmitz Jr., 1994, "Can Companies Maintain Their Initial Innovation Thrust? A Study of the PC Software Industry," *Review of Economics and Statistics,* 76: 523-540.

[13] Stavins, J. 1995, "Model Entry and Exit in a Differentiated-Product Industry: The Personal Computer Market," *Review of Economics and Statistics*, Volume 77, pages 571-584.

[14] Steffens, J., 1994. Newgames: Strategic Competition in the PC Revolution, New York: Pergamon Press.

[15] Sims, C., 1980, "Macroeconomics and Reality," *Econometrica,* 48: 1-48.

List of Footnotes

1. See Gabel [1991], Bresnahan and Greenstein [1997b], Langlois and Robertson [1992], Ferguson and Morris [1993], Cringely [1992], Steffens [1994].

2. This comparison is sometimes inappropriately applied to the earlier time period since "closed system" best describes a later Apple product, the MacIntosh, which was launched in 1984-85.

3. In section 3, we present econometric evidence that the development of the Apple platform was quite different from that of the two open platforms.

4. At its introduction, the IBM PC also could run the CP/M operating system.

5. This 16 bit version of CP/M was released just before the introduction of the IBM PC.

6. Interestingly, Gabel also provides evidence that software applications for the Apple operating system declined significantly with the advent of the IBM PC and the failure of the Apple III to catch on.

7. It is well known that IBM attempted to encourage peripheral and software development.

8. IBM's relationship with the business community & software providers cannot be the only reason for the success of the IBM PC. DEC which had a large installed base of minicomputers tried a strategy similar to that employed by IBM. Its foray into the personal computer market failed, even though the DEC Rainbow could run both DOS and CP/M operating systems. Additionally, even as late as 1983, there was widespread dissatisfaction among Data Processing managers with functionality of personal computers in business environments (Friedman and Cornford [1989]).

9. Peripherals include all non-software accessories that did not come bundled with the computer itself. This category includes modems, floppy and hard drives, back-up devices, graphics cards, etc.

10. The figure for peripherals is similar to that of software.

11. Of course actual software sales would probably be a better proxy for the availability of complementary products. Such extensive data were never compiled by any of the major industry trade publications. Prusa and Schmitz [1994] collected data on software sales for some of the years included in our study, but their data does not include all categories of software.

12. Complete sales data for microcomputers do not exist. Perhaps the closest is the data collected by Stavins [1992], which has sales for a majority of the popular microcomputer models, but not the entire market.

13. Despite economists' general interest in the phenomena of advertising, as a research topic in itself, there is almost no precedent for using this type of data to learn about features of the underlying high technology market. We are aware of only one other attempt to examine advertising in high- technology markets. Klenow [1994], uses news releases and announcements to track the entry of new goods.

14. While there was advertising for other proprietary platforms in this period, notably TRS, and Atari, these are less interesting. First, they are quantitatively less important. Second, our impression is that TRS and Atari were primarily targeted for the "games" market, i.e., a different set of consumers. Figures 1 and 2 confirm that there was relatively little advertising for these two platforms in Byte.

15. Indeed the CP/M ratios of software/hardware and peripherals/hardware system advertising were biased upwards; at the time of CP/M's death there was almost no advertising for microcomputers running the CP/M operating system, while there was still plenty of software and peripheral advertising.

16. These supplementary regressions are available at the JIE editorial web site (http://haas.berkeley.edu/˜ jindec).

17. The VARs with a single lag are available at the JIE Editorial web site. In some cases, we could not reject the hypothesis for some specifications that the VAR was only a single lag process. We show all the two lag estimates for a sense of symmetry across specifications. Generally speaking, there are insufficient number of observations here to identify a third lag at any reasonable statistical significance. Similarly, there are not enough observations to identify a third variable (such as peripherals) in a two-lag VAR. Note that for DOS, table (II) still has 22 observations, despite the two lags. This is because advertising for products running the DOS platform began in the third quarter of 1981, that is, there is no DOS advertising for the first two quarters of that year. Since DOS has so few observations to begin with, we use that information in order to have 22 observations for DOS.

18. If c is known, from (4), the constant k can be found by setting $A(L)k = c$.

19. δ_2 and δ_3 are non- linear in α and β. The standard deviations in this table were calculated using the *Delta Method.* See Greene (1993), p.297 for details.

20. Table III show that only the second lags (the δ_2's) are significantly different for DOS and CP/M when the operating system is the impulse and software is the response. From Table III, a 90 percent confidence for δ_2^{dos} is (0.68, 2.12) and a 90 percent confidence interval for δ_2^{cpm} is (-0.05, .55).

Notes to the tables:

- Recall that advertisements for microcomputers running the relevant operating system are our proxy for operating system sales; in the tables we use the abbreviation OS for this variable.

- In the VAR regressions, the numbers in parentheses in table 2 are the standard errors. A "*" means that the t-stat exceeds 1.64, a "**" means that the t-stat exceeds 1.96.

Category	*Mean*	*Std.Dev.*	*Minimum*	*Maximum*
OS CP/M	7.28	6.43	0.00	20.00
Soft CP/M	11.54	7.74	0.50	27.50
Periph CP/M	3.60	2.79	0.00	10.00
OS DOS	11.75	8.75	0.00	29.50
Soft DOS	25.02	17.56	0.00	62.00
Periph DOS	22.11	11.32	0.00	38.00

Table I: CP/M: April 1978:1 1986:4 (36 obs.); DOS: 1981:3 - 1986:4 (22 obs.)

	Dependent Variable			
	OS (DOS)	OS (CP/M)	SOFT (DOS)	SOFT (CP/M)
Independent Variables				
constant	1.67	-0.71	3.48	2.06*
	(1.34)	(0.95)	(2.81)	(1.06)
OS (-1)	-0.28	0.39*	0.85*	0.52**
	(0.24)	(0.20)	(0.51)	(0.22)
OS (-2)	-0.017	0.09	0.90**	-0.0077
	(0.21)	(0.19)	(0.44)	(0.22)
Soft (-1)	0.47**	0.39**	0.58**	0.66**
	(0.14)	(0.17)	(0.29)	(0.20)
Soft (-2)	0.11	-0.0026	-0.45	-0.17
	(0.18)	(0.17)	(0.38)	(0.19)
Adj R^2	0.82	0.82	0.80	0.84
DW	2.05	1.99	1.94	2.03
Number of Obs.	22	34	22	34

Table II: VARs: Two Lags: IBM/DOS (1981:3 - 1986:4) & CP/M (1978:1 - 1986:4)

Response	Impulse	Impulse Response Parameters		
		δ_1	δ_2	δ_3
OS (DOS)	Software (DOS)	0.47	-0.022	-0.020
		(0.14)	(0.12)	(0.11)
OS (CP/M)	Software (CP/M)	0.39	0.15	0.093
		(0.20)	(0.13)	(0.04)
Software (DOS)	OS (DOS)	**0.85**	**1.40**	**0.43**
		(0.51)	(0.44)	(0.43)
Software (CP/M)	OS (CP/M)	0.39	0.25	0.18
		(0.17)	(0.18)	(0.10)

Table III: Impulse Response Parameters (Directly Calculated from VARs)

	APPLE	
	Dependent Variable	
	Hard (APPLE)	SOFT (APPLE)
Independent Variables		
constant	0.57	0.17
	(0.63)	(1.07)
Hard (−1)	0.24	0.21
	(0.18)	(0.32)
Hard (−2)	0.36*	0.53
	(0.19)	(0.33)
Soft (−1)	0.020	0.35**
	(0.10)	(0.17)
Soft (−2)	0.039	0.33**
	(0.093)	(0.16)
Adj R^2	0.21	0.62
DW	1.82	1.91
Number of Obs.	34	34

Table IV: VARs: Two Lags: Apple (1978:1 - 1986:4)