Lock-In and the Costs of Switching Mainframe Computer Vendors: What Do Buyers See?

Shane M. Greenstein
Department of Economics
Lock-In and the Costs of Switching Mainframe Computer Vendors: What Do Buyers See?

Shane M. Greenstein
Department of Economics
University of Illinois at Urbana-Champaign

This essay is a substantial revision of the second chapter of my Stanford doctoral dissertation. Tim Bresnahan, Paul David, and Roger Noll provided more sagacious advice than I could possibly follow. Discussions with Pablo Spiller have been very useful. The center for Economic Policy research at Stanford University, The Babbage Institute, and the National Science Foundation provided funding. All remaining errors are my own.
Abstract

No careful empirical research has confronted the widely-held belief that the costs of switching computer vendors tends to produce technological "lock-in". Using several studies by federal agencies into the costs of switching mainframe computer vendors, this article concludes that the present literature overlooks many important factors that lead a computer buyer to lock-in to an incumbent vendor. For one, a computer user can invest in (far-sighted) "anticipatory converters" in advance of a future acquisition. These converters reduce the realized level of anticipated switching costs. In the cases studied here, the incentives for making these investments depended on many factors, and the incentives were usually too low. Second, the level of future switching costs are often not known to the decision-maker when a potential vendor switch is considered. This is important because managers can bias their estimates of the costs of switching vendors and, in effect, influence the extent to which a procurement favors an incumbent. In the cases studied here, because the estimates of switching costs were often subject to large errors, and because the costs of making a wrong estimate fell largely on computer users, users had incentives to bias their estimates to favor incumbents. Finally, this experience suggests that analysis of switching costs should not treat buyers as monolithic organizations, i.e. decisions pertaining to computer system use, switching cost estimation, and vendor selection were not coordinated within the federal bureaucracy. Rather, resolution of bureaucratic conflict influenced tendencies for users to "lock-in" to an incumbent vendor.
In most cases new ADP (Automatic Data Processing) technology will require modifications in system configurations, telecommunications and especially software, that can become intricate, lengthy and difficult to resolve. Hence,... managers in both the public and private sectors tend to prefer new technology that is as compatible as possible with existing technology to minimize disruption in the conversion process.


I. Introduction

Switching costs are costs incurred as a consequence of a buyer switching between alternative suppliers of essentially the same product. Large switching costs can make buyers reluctant to switch suppliers. This reluctance has two important consequences. First, it potentially provides incumbent suppliers market power.1 Second, it may influence buyer and supplier choices among alternative technologies for a product. Markets may "lock-in" technical alternatives compatible with early technological leaders and "lock-out" incompatible alternatives.2

Notable empirical studies of technology lock-in include studies of nuclear power plant design (Cowan 1988), video cassette recorders (Rosenbloom 1989), the typewriter keyboard (David 1985, 1986), and stereo systems (Postrel 1990). These studies show that if a user invests in systems of compatible components, some past investments typically retain their value when the user purchases more compatible components but lose their value when the user purchases incompatible components. This produces an inter-temporal link in a buyer's decisions, which, in turn, influences producer pricing, output and product design decisions over time. All these intertemporal links help produce technological lock-in.

The computer market is another example often cited as one that fits the mold. Because users of computer systems invest in systems of compatible components, i.e. hardware and software, it is widely believed that computer users tend to "lock-in" to their incumbent vendors.

No careful examination of the computer market has confronted this belief with empirical fact nor with skepticism. This is unfortunate because the belief need not necessarily survive the close scrutiny of such a
confrontation. For example, that buyers invest in a systems of compatible components, by itself, does not establish that buyers find it costly to switch vendors relative to the benefits of doing so. Nor, by itself, does it establish that buyer and vendor decisions are linked in a way that produces technological lock-in.

Why has this belief not been examined? The absence of empirical research on this issue is probably best explained by the difficulty of collecting information about the majority of computer users in the United States, who are private firms. Summaries of many users' experiences with changing vendors, if there is any experience at all, are usually not publicly available. What is often made public, surveys of "customer loyalty," provides ambiguous inferences regarding the relevance of switching costs to most buyer's decisions.\(^3\)

Because of the dearth of information about private firm's experiences, it would be remarkable if enough examples existed for an empirical study of technological lock-in. One of the contributions of this article is that it brings one such set of examples to light.\(^4\)

The set of examples used in this article were recorded in the late 1970s. At that time many federal agencies began to experience large expenses related to the conversion of their software from one mainframe architecture to another. These "conversion costs" raised a number of unexpected problems when agencies ordered replacement acquisitions. Because of these problems, the topic of conversion was closely studied by several federal oversight agencies who employed specialists in the computer field - i.e. the Government Accounting Office (GAO), the National Bureau of Standards (NBS), and the General Services Administration (GSA).\(^5\) These studies left a detailed and public record of the conditions surrounding more than a dozen actual acquisitions and the solutions attempted to conversion problems.

The analysis of this article aims to synthesize these studies for an audience interested in the economics of lock-in. The analysis will identify the structural conditions contributing to switching costs and lock-in, if any occurred at all in practice. The principal novelty of this article derives from its conclusions. While the analysis does not conclude that the
present literature users improper "stylized facts," it does argue that the present literature has overlooked many relevant structural features that determine a buyer's tendency to lock-in to an incumbent vendor. In particular, this article strongly questions the fundamental presumption of previous analysis that buyers can costlessly formulate and implement consistent inter-temporal strategies for dealing with high vendor switching costs.

II. A Summary of the Analysis

With the notable exception of Williamson's (1975, 1979) pioneering work on related issues, the present literature on switching costs has generally treated users as monolithic, overlooking the role of decision-making at many levels within a organization making a purchase. In contrast, the studies done of federal agencies demonstrate that the structure of decision-making within an organization determines the incentives of a buyer to lock-in to an incumbent vendor.

This argument is broken into four sections. The first section verifies what was long suspected: mainframe computers systems possess many of the technical features that potentially lead to large switching costs. That is, computer systems exhibit "technological interdependence" and incumbent vendors possess advantages in a competitive procurement. The first section also summarizes the suggestive, but inconclusive, empirical evidence on the magnitude and importance of switching costs for vendor choice. The ambiguity of these conclusions motivates a close examination of acquisitions in practice, a task that constitutes the bulk of the analysis.

The second section discusses how difficult it is to estimate the costs of switching vendors. In practice, buyers make estimates that are often subject to large errors. Moreover, buyers have only a limited number of options for reducing those errors. That is, the price and quality of a non-incumbent's system has uncertainty associated with it. This is problematic since a large underestimate of the costs of switching vendors can lead to a costly switch to a nonincumbent vendor, while a large overestimate grants excessive market power to an incumbent providing an upgrade. This dilemma influences how estimates are made (and biased) and how the costs of being wrong are allocated, which, in turn, influences the extent to which an
incumbent vendor is favored.

The third section examines whether switching costs are outside the control of the user. It finds that vendors' technology choices were not solely responsible for a buyer's switching costs, as is generally presumed by most previous economic analysis. Rather, users can invest in "anticipatory converters". That is, in anticipation of a future acquisition, users could make (far-sighted) investments that could significantly change the level of switching costs later. The analysis argues that the incentives for making these investments depend on many factors within an organization, but were generally too low in the cases studied here.

The final section highlights the relevance of bureaucratic conflict for the vendor choices made in practice. As was common for mainframe purchases throughout the country in the 1970s, the buyer and the user were usually not the same decision-maker -- i.e., large capital purchases like mainframe computers were funded by someone other than those who eventually programmed the system. However, those who provided the funding often disagreed with those who used the system over decisions pertaining to vendor choice and system use. Moreover, those who funded the purchase could not easily monitor the actions of those who used it. In practice, resolution to this bureaucratic conflict largely determined the tendency for users to "lock-in" to an incumbent vendor or not, because it determined who provided the estimates of the costs of switching vendors and how the burden of those costs were funded. These decisions, in turn, influenced the investment in anticipatory converters and the importance of switching costs for an actual vendor purchase.

The main danger in studying federal agencies is that their experiences may be unrepresentative of other mainframe users in the country. Though all expenditure on computers by public and quasi-public organizations represents roughly a quarter of the United States computer market, federal agencies comprise less than 5 percent of the mainframe market. To guard against the danger of inference from an odd example, the discussion focuses on agency use of commercial general purpose computers commonly used in the United States in the late 1970s and early 1980s. Moreover, the discussion highlights where the federal procurement process resembles or differs
significantly from other mainframe users, universities and private industry.

III. Switching costs and mainframe computers in federal agencies

This section establishes that mainframe computers in the federal government do possess the technical features usually associated with technical lock-in.

In most models of buyer choice subject to switching costs, a buyer makes investments with a supplier that continue to have positive value if the same supplier is selected again, but have no value if any other supplier is chosen. The motivating case is one where an old and depreciated system is replaced with a new similar system. Some old equipment works with a new system from the incumbent supplier, but not with equipment from any other supplier. It is as if the nonincumbent supplier charges a price, $P_{ni}$, for his system and the buyer pays that price, while the incumbent charges $P_i - S$ for essentially the same system, where $S$ is the value of the old equipment the buyer will continue to use with the new system.

The simplest theoretical representation seems to describe switching mainframe vendors, though it misses some other (one time) expenses associated with switching. In practice, switching costs arise due to: (a) site preparation, such as raising floors, installing cooling units and electrical and communication connections; (b) training personnel to use a new vendor's unique system features; (c) dual operation of systems while one is installed, tested, and brought up to an acceptable operational level; (d) disruption of operations while new hardware is installed; (e) re-optimizing new systems to unanticipated problems; and (f) the direct costs of software conversion if existing software was a component worth preserving. Most of these expenses are minimal if one stays with the same supplier when upgrading mainframe purchases, but the costs of (b), (c), (e), and especially (f) can be substantial if a change between suppliers of incompatible technologies occurs.  

a. Technical interrelatedness and incompatibility.

Previous work has identified several technical features of products that result in switching costs, and, not surprisingly, these features can
be found here. First, commercial computer systems display what Paul David (1975, 1985, 1986, 1987) has called "technical interrelatedness": tasks require a multi-component system and that set of components must be technically compatible in order to work together and achieve efficiency in system performance.

It is easy to illustrate each of the two aspects of technical interrelatedness with mainframe computers. An essential technical feature of general purpose computer systems is that they are composed of a variety of compatible components -- central processor unit(s), input-output devices, communication terminals, memory devices, and "software" -- the programming that directs a system's hardware to manipulate information in a desired predictable manner. Table 1 contains a listing of the number of these components for different types of commercially available general purpose computers in use by federal agencies in 1979, clearly demonstrating that a large number of a variety of components make up a typical commercially available general purpose system. Table 2 illustrates that these components are typically supplied by the same firm. Note, however, that though hardware equipment complimentarities are important, the most important complementarity for economic purposes is not displayed -- the relationship between hardware and software, and/or software and human training.

Evidence of the compatibility requirement is also easy to find, especially in the extensive discussions of the cases when that requirement is not satisfied. Examples of incompatibilities include the following: plug and socket do not necessarily fit together physically or if they do fit together, they may not identically translate electronic signals in a similar manner; system software, the code that translates user written code into machine commands, is usually unable to work with hardware architectures other than the one on which it is written unless the software is altered; higher level software, usually embodying commands for a particular application, can be incompatible with particular system software implementations available on the new machines; higher level software that is optimized for implementation on one machine architecture can lose significant performance if implemented on another system. In this case compatibility is not a dichotomous variable, but rather, describes a condition that varies by degrees; data files written in a particular form
may be unsuitable for specific hardware models (NBS 1980); different manufacturer's systems use different system software, requiring retraining of the personnel who use the system.

Because of all these levels of technical interrelatedness, switching costs arose due to changing from one incompatible systems to another, not due to changing suppliers, per se. However, because there tended to be a one-to-one association of suppliers with computer mainframe technical families in the 1970 and early 80s, with a few notable exceptions, these differences tended to be indistinguishable in practice. In general, users could not "mix and match" components from different manufacturers. This well-known feature of systems was frustrating to users.

b. An incumbent's advantages.

With vendor's offering incompatible systems, two factors worked to an incumbent's advantage in a competitive procurement to replace hardware. Long-lived assets obviously give the incumbent an advantage. Different components depreciate at different rates, leaving, at any time, some components that could continue to be used in future systems and some that required immediate scrapping. For example, the median age of processors of commercially available general purpose systems in 1979 was six years (mean = 6.3, s.d. = 3.6). In addition, software and programming obviously does not physically fall apart, nor is previous training immediately forgotten. Once operational, software and training are useful as long as the proper complementary components are in place. Hence, to provide an equivalent, fully operating system the nonincumbent must supply features which the incumbent need not.

Focusing on only length of life of assets paints a somewhat deceptive picture, however. Also important is whether that old hardware, software, or training continues to have value in use. The value in use varies for each situation, depending on the demand for services the system provided and the necessity for reconfiguring a system to provide a new application. Generalizations are difficult because the details vary from one situation to another. In some instances, the interrelationship between the idiosyncracies of an application and the idiosyncracies of software constrained changes in the applications, which left the incumbent as the natural supplier of complementary components. In other cases, where the
applications were in need of radical alteration, rewriting most of the system software was a viable economic alternative.

A second factor contributing to an incumbent's advantage in a competitive procurement involves the time it took to correct conversion problems. It is well known that many improvements in software on the incumbent machine -- improvements that were added over time and through much use and testing -- cannot be easily translated to a new machine but, in fact, must be reinvented through extensive testing and trial and error aimed at learning how to take advantage of the unique features of the new system. Moreover, this trial and error must occur in sequence -- only after one part of the program is polished can another be developed. In addition, programmers cannot usually anticipate all the practical difficulties to achieving a desired "look and feel" in system performance. Agency's experiences confirm this general property of software design: no other input, even a very elastic supply of programmers, can substitute for the amount of time needed to refine software through trial and error.¹⁶

The consequence of this technical constraint should not be underestimated. Users faced a choice between (a) a relatively "quick" conversion to a compatible system, or (b) a longer wait (as much as a year or more) to install a new working system from a nonincumbent. More will be said about this later.

c. Observing switching and its determinants.

Establishing the existence of switching costs, by itself, does not demonstrate their relevance over a wide range of situations. Switching costs will be relatively larger in some situations than in others. Switching costs matter most when old hardware components, particularly processors, are to be replaced with new technical generations because this situation requires that all the software on the old system be made compatible with the new system. Switching costs also influence supplier choice if existing systems and new systems must work together in the new configuration, or if the new system will employ any software developed on the old system. Switching costs, in the sense used earlier, are not relevant to vendor selection where old technologies have been orphaned -- e.g. no new upgrade is available, which was not uncommon in the 1970s. Incumbents no longer possess advantages because the old equipment is not
upward compatible with any new generation. All potential vendors and the incumbent are on equal footing. However, an orphaned system's switching costs may influence other aspects of new system choice, such as the timing of an acquisition.  

What is the quantitative evidence that the switching costs were a large problems in many cases? Despite much information about computer procurement, an answer is difficult to formulate because the relative influence of switching costs on buyer decisions are not easy to measure, and disclosure laws prevent outsiders from examining the costs that influenced decision-making in an actual acquisition.

However, several publicly available studies were done of federal agency conversions. These are summarized in the appendix to Greenstein (1989). These studies found that the total switching costs between incompatible systems could vary over a large range. They could be anywhere from 22 to 250 percent of the price of the acquisition of the new system, depending on management practices, software quality, and other uncertain factors. By another measure, these conversion costs were anywhere from 13 to 128 times the average monthly rental for the newly acquired system (or roughly one to ten times the average yearly cost). These studies leave the impression that switching costs are distributed asymmetrically — bounded from below and skewed rightward by a few especially costly unpredictable circumstances.

The lowest conversion expenses involved an upgrade between machines from the compatible IBM system 360 and 370 families. All the system-specific features of software implemented on the 360 were preserved in the upgrade to the larger 370 machine. These conversion expenses totalled 1 percent of price of acquisition or one-half the costs of one month's average rental.

There are several reasons to think that conversion expenses in federal agencies could be large. The installed stock of commercial general purpose mainframes, as shown in Table 3 is quite large, often exceeding 2,000 systems in any year. Table 4 shows that the number of new acquisitions per year is also high enough that if switching costs are a problem it is likely that the dollar value of this problem is large for the whole federal
government. Finally, Table 5 lists the publications by federal oversight agencies that dealt with problems related to switching vendors. It is hard to imagine that this effort would be expended if switching costs were not a problem for a substantial number of federal agencies.

Despite the previously described suggestive evidence, systematic econometric evidence has led to ambiguous conclusions. The main observable consequence of switching costs -- repeated buyer choice of the same product -- may also be explained by persistent buyer preferences for the unique services provided by a vendor. Rarely does a set of micro-level data provide direct measures of either a customer's anticipated switching costs or buyer preferences; thus data does not permit an observer to distinguish between the effects of switching costs and others preferences in accounting for temporal patterns of purchases. Econometricians have not found methods for solving this problem other than imposing arbitrary functional forms.22

Despite these difficulties, Greenstein (1991) was able to find some evidence that incompatibility influenced vendor choice from 1972 through 1983. This research investigated the patterns of choices by offices of federal agencies that had experience with only one vendor prior to their next acquisition. It found that after many economic factors were controlled for, a large part of the tendency for former IBM users to switch to other vendors arose primarily from the fact that limited compatible upgrades were available for users of very old IBM equipment (e.g., the IBM 1400 series). Federal buyers who previously used IBM equipment and could upgrade to a new compatible system (e.g. IBM 360/370) tended to choose IBM as often as users of other vendors selected those vendors again. This result is the first econometric analysis of the relevance of compatibility for vendor choice.23

Other quantitative work yields more cautious conclusions. Greenstein (1990) investigated whether the extent of a user's investment, which should correlate with switching costs, helps predict whether agencies sole-sourced, rather than competitively procured, from an incumbent. This research formally tested the hypothesis that high switching costs could give incumbents such an advantage that agencies would have little reason to solicit multiple bidders under competitive procurement procedures. From a sample of single-incumbent users, it found that the extent of investment with an incumbent does positively predict sole-sourcing from the incumbent,
but it does not dominate the observed results, contrary to expectations. Many other economic factors predicted the choice of procurement procedures and can often overwhelm any apparent (and measured) advantage an incumbent seems to have.

Quantitative evidence is suggestive, but inconclusive. A deeper understanding of switching costs warrants careful analysis of actual acquisitions. The following analysis accomplishes this by analyzing the problems confronted by federal agencies who switched to a new mainframe vendor for a replacement acquisition.

IV. Two additional factors

The following section analyzes two factors that play important roles in the case studies of switching costs: (1) the difficulty of estimating future switching costs; and (2) the extent to which switching costs are a function of previous user behavior. To emphasize that these factors are distinctive from organizational issues, the focus of the last section, and to provide a basis for more general conclusion, the discussion focuses on agencies that use commercial computer technology and does not rely on features peculiar to the computer's use in the federal government.

a. Uncertainty about switching costs.

Previous theoretical work has closely analyzed the consequences of one source of uncertainty: buyers make their initial investment in a vendor's system without knowing who will have the best system for their needs in the future (e.g. Klemperer, 1987). This is clearly a factor in vendor choice, though perhaps, less so in federal agencies where the choice is often done in a formal procurement process that cannot easily accommodate long-term expectations about industry trends (Kelman 1990).24

Rather, this case highlights the relevance of other sources of uncertainty. In particular, when buyers choose between incumbent and non-incumbent vendors, they typically do not know with much certainty the likely future monetary level of switching costs, the overall feasibility of conversions, nor the likely length of time to complete a conversion. Many of these problems can be traced to unanticipated and unavoidable problems in "fine-tuning" newly converted software. Moreover, how these uncertain
variables are estimated is important for future vendor choice. A large underestimate of the costs of switching vendors could lead to (ex post) an "unnecessary" and costly switch to a new vendor, while a large overestimate effectively grants an incumbent vendor monopoly power over the next upgrade, which can lead to a more costly acquisition price.  

There was substantial technical uncertainty surrounding the feasibility of conversion. It was difficult to "transport" software between incompatible systems, because software typically embodied features needed for a unique application in the agency and was technically complementary to the system on which it was developed. In practice, software was written for a particular set of needs and for a fixed set of users trained to use it. Software lost some (if not all) of its functions when implemented on alternative systems, even those that were technically more advanced, as measured by benchmark programs. Hence, it was difficult to duplicate easily a former system's performance using the same software on another system.

Agencies could anticipate that there would be problems during conversion, even if they could not anticipate what those problems would be. Large conversion expenses were inherently difficult to estimate, even for experienced conversion experts. Software conversion costs did not follow a calculable algorithm based on a readily observable feature of the code, such as the number of lines. Unanticipated costs could often be traced to poor documentation of earlier programs, fragile code -- held together by "bubble gum and bobby-pins" -- which was difficult to get working again until a crucial bottleneck in the code was understood, and "patchwork code" -- a program composed of unsystematic additions to the basic software program, the logic of which was hard to reconstruct years after the program's many creators had departed from the agency. Any one of these characteristics made it difficult to retrace the operational logic underlying old programs. Many of these problems were difficult to anticipate until software conversion was underway or largely accomplished.

In the 1970s there was little an agency could do to reduce that uncertainty. During a conversion agencies could either invest in preserving old software on new machines or invest in reinventing their software on the new hardware. Either option was time-consuming and costly. In-house
conversions usually took too long because the required number of programmers exceeded an agency's available staff, especially with large jobs. Moreover, old staff usually had little experience with conversion and misunderstood what was required. Programmer knowledge about software implementation and programming procedures were useful on an old system but not necessarily useful on a new system. It was no better out of house: contracting out for conversion services could be quite difficult and expensive because performance standards were difficult to specify in a contract, especially when the output was idiosyncratic. Conversion experts also were difficult to find, because this type of problem was not common in private industry. Many private firms were undependable, and agencies frequently had to use their own staff to refine the conversion programs for which they contracted.

Not surprisingly, conversion costs could be greatly underestimated or overestimated if the agency's office was not very experienced with conversion, which they frequently were not (since conversions occur infrequently at the same location). Conversions could also be greatly underestimated or overestimated if the conversion work contained several unpredictable and largely intractable problems.

All of this was problematic since a large underestimate of the costs of switching vendors could lead to a costly switch to a new vendor, while a large overestimate effectively grants an incumbent vendor monopoly power over the next upgrade, which led to a more costly acquisition price. Not surprisingly, reduction of the uncertainty surrounding conversions became the focus of many publications and efforts. For example, since the early 1980s, the Office of Software Development in the GSA has housed experts in conversion problems, professionals who are experienced in the special tools required for these problems. There is also considerable documentation of oversight and advisory agencies providing aid in the form of expert advice, bibliographic material on conversion tools, and other managerial guidance material. For evidence of this, see Table 5. Ironically, these effort to extinguish the blaze are the best evidence that the blaze was large.

In sum, buyers had limited options for reducing the uncertainty over a nonincumbent's product quality and eventual costs of installation. Not only could switching costs be large, but a nonincumbent's eventual "price" was
subject to a large variance.

Generally speaking, it is important to understand the incentives of managers and users to bias their estimates too high or too low. For example, to the extent that decision-makers incurred some of these costs and to the extent that they were risk-averse, this enhanced the advantage an incumbent vendor already possessed. The analysis below will argue that how the risks of being wrong were allocated between buyer and seller influenced the incentives of managers and users to bias their estimates. These estimates, in turn, largely determined the actual vendor choice.

b. Converter technologies and anticipatory investments.

A second feature of switching costs highlighted by this case is the extent to which buyers control the level of future switching costs. In most models of markets subject to switching costs, buyers are assumed to make a vendor/technology choice, then use the product for some time and finally make another vendor choice, where the later choice may be subject to an exogenous level of switching costs. While this partially resembles the situations observed in federal use of mainframes, it misses important links between the decisions made about the use of a system and the costs paid to switch vendors at a later date.

That users influence the level of future switching costs to some extent is not too surprising, especially if decisions are coordinated over time. When decisions are coordinated, buyers can spread switching costs over time in innocuous ways. After all, if a user can foresee with certainty that he will be staying with the same hardware vendor in the future, he will likely telescope that decision to the present by taking anticipatory actions. For example, buyers may purchase incremental peripheral equipment for existing systems or develop software that raises switching costs (were a switch to be made). Such actions are irrelevant if under all circumstances the next system upgrade is with a compatible vendor.

In fact, the links between user actions and future switching costs are more complex than that. Users can make extensive efforts to change future switching costs, and those efforts are often related to decisions made daily. In the case of federal computer systems, these efforts included
standardization of component parts, greater use of higher-level language programming, and efforts to achieve modularity of software and system design and structured programming. Two users, starting with exactly the same system, could end up with substantially different switching costs if their tendency to make anticipatory investments differed, i.e. if their day-to-day use and programming practices largely differed.

The analysis of these activities can best be placed in the context of a discussion of "converter" technologies -- bridges between incompatible systems that free the buyer to use alternative system sub-components without necessitating investment in an entirely new system. Converters have received some attention in the literature because their introduction has important (and sometimes surprising) consequences for industry dynamics.\(^{33}\) It is now recognized that third parties can enter with technical "bridges" to system incompatibilities. Those converters result in systems that are complementary in various sub-components, thereby integrating systems at various costs.

Instead of focusing on converters provided by third parties, this case highlights the role of converters provided by users. Here two types of converter technologies were important in mainframes -- anticipatory and retrospective converters, a distinction previous research has not recognized. Retrospective converters are tools for easing the pain of conversions when they take place -- often provided by parties other than the buyer or system vendor. Anticipatory converters differ in that they are installed by the buyer prior to any definite decision to switch suppliers. For technical reasons explained later, anticipatory actions regarding mainframe computers can only be taken prior to the decision to go through with a conversion.

Examples of tools for retrospective conversions can be found in the Office of Software Development in the GSA, as noted earlier. This office's existence alone reveals the benefits of trying to ease conversion costs once a decision to convert has been made. In addition, bringing old software installed on new machines up to performance levels achieved on the old system took time and manpower, a cost agencies willingly incurred to save software. There was also an additional opportunity cost associated with taking programmers away from their efforts to improve the performance
of existing programs.

Since retrospective conversions were costly, yet desired, it would not be surprising to observe farsighted actions designed to reduce the costs of anticipated retrospective conversions, should that option be likely. As predicted, examples of attempts to use anticipatory converters are seen in the attempts to make all software "transportable" prior to any switch -- i.e. make it perform consistently when implemented on technically comparable (or improved) new systems possessing architectures (or system software) incompatible with the one on which the software was originally developed.

Attempts occurred in both government-wide programs and at the level of an agency's office. Some government wide programs passed the costs of anticipatory conversions to vendors. These included attempts to standardize manufacturer's higher level languages, and attempts to coordinate manufacturers to produce similar physical interfaces. Other efforts aimed to share the costs of anticipatory conversions among agencies included efforts to standardize software at different agencies on a few well-developed programs and efforts to establish software pools, where agencies can swap programs. Aside from eliminating redundancy, the latter two efforts tried to make basic software available to all agencies, no matter who the hardware manufacturer was. Still other government-wide efforts attempted to change agency programming. These were connected with attempts to standardize all programming in higher level languages, such as ADA or Cobol, and attempts to provide advisory material on the need for "documented, modular programming" in higher level languages.

All these efforts, if followed, were designed to result in systems that were composed of interchangeable component parts. These actions attempted to make conversion a more routine procedure, eliminate some of the uncertainty about the magnitude of switching costs, and reduce some of the need for retrospective converters. The goal was to have a system of code that would perform on any manufacturer's system at any time, regardless of the one for which it originally was developed and of when it was developed. This is precisely what we would expect an investor anticipating conversion problems would try to do.
Such anticipatory converters were costly to do; and it may have required programmer cooperation among many agencies, which was not easily forthcoming. Often there were differences between the organization's and agent's benefits from taking these actions, a topic discussed later. Whatever the cause, case studies of conversions in the mid 1970s make clear that ideal programming practices were generally not followed in the past. Most agencies' stocks of software remained compatible with a limited set of available architectures at any point throughout the 1970s.37

In sum, previous user management decisions influenced the costs of switching vendors when a switch was made. Users who did not make efforts to install anticipatory converters faced higher switching costs than those who did. Hence, if conversions are a likely option, it is important to understand the incentives of managers to account for the value of making efforts to reduce their cost.

V. "Lock-in" within Bureaucratic Organizations

The existing theoretical literature on switching costs often presumes that the level of costs is exogenous to the buyer and known at the time of purchase. The previous discussion showed that these assumptions overlook two other characteristics of switching costs in practice, that switching costs are uncertain and are partially endogenous. This section discusses several more reasons why this is important. Uncertainty and the endogeneity of switching costs interact with bureaucratic conflict, either increasing or decreasing tendencies for buyers to lock-in to incumbent vendors or incumbent technology.38

The key distinctions in this section are between the "user" of a system, the "buyer" of a system, and the "estimator" of the costs of switching vendors. The analysis has not needed these distinctions until now. As shown below, these cases demonstrate that the same individual need not have responsibility for daily system use, choices among competing vendors, nor evaluating the costs of switching vendors. When decision making for these three decisions is split among different individuals, related decisions do not tend to be well-coordinated. Thus, the assignment of responsibility for decisions and their coordination, determines, in effect, the ultimate importance of switching costs for vendor choice.
a. The Separation of Funding from Use

As was common for mainframe use throughout the country in the 1970s, in the federal bureaucracy large capital purchases like mainframe computers tended to be funded by someone other than those who eventually used the system. This separation of funding and use often lead to principal/agent conflicts over vendor choice and generally over many parts of the procurement process. In this case, the "principal" was the US Congress or its designated oversight agency, who ultimately provided funding for the equipment. The "agents" were those within the federal agency who used the mainframe computer and who ultimately invested in anticipatory converters. Quite typically a small number of experts within the computer systems department of the agency made the majority of the daily decisions concerning mainframe use and the "principal" could monitor these decisions only imperfectly.

The heart of the problem is that even if the principal and the agent both used the same information, they are likely to perceive different benefits and costs from choosing an incumbent vendor instead of a non-incumbent vendor for a replacement acquisition. In the general case, it is commonly believed that agent's value technically proficient systems more than principals and that agents do not value dollar savings as much as the principal. In other words, an agent will prefer the computer equivalent of a "Cadillac", while the principal would rather he used a "Chevy" (Kelman, 1990, Marshall et. al., 1990, Greenstein, 1991).

Switching costs complicate this conflict because the costs to the principal of switching are not likely to be the same as to the user. This observation has two parts. First, the principal pays the costs of switching no matter which vendor wins a bid. That is, switching costs grant the incumbent some monopoly power, so he can bid less aggressively. If the incumbent wins the procurement, the principal ultimately pays a high price to the incumbent. If the nonincumbent wins, the principal ultimately pays for the costs of switching to the nonincumbent. Second, for the agency, the costs of the two outcomes can differ substantially. Switching to a nonincumbent takes time and effort, and it prevents system users from pursuing other projects, private opportunity costs the capital budget generally only partially covers. Staying with the incumbent costs the
agency much less in these terms. Thus, the agent generally has less incentive to switch vendors than the principal.

This principal/agent conflict turns the estimation of switching costs into an extension of the general bureaucratic conflict. Users would like those estimates to reflect the user's private costs of switching, while the principal would prefer that those estimates reflect the entire organization's costs and benefits. The principal will likely distrust any estimate of switching costs made by users, since the user has an incentive to exaggerate the costs of switching vendors as a means to favor the incumbent. Due to their experience with the system, however, users are better positioned to estimate the likely costs of switching.

The principal/agent problem also turns the effort expended to reduce the costs of anticipated switching vendors into an extension of the general bureaucratic conflict. This is not so surprising since, as was shown in the previous discussion, there is no obvious correspondence between an user's private costs and benefits of expending effort and the organization's costs and benefits. In general, it is widely believed that the programmer's private incentives are too low. This belief is partly due to the positive externalities of standardization: it takes effort by one programmer, but many other users may benefit. This belief is also partly due to historical experience -- e.g. the absence of ideal programming practices in the 1970s, as already mentioned.

In sum, the assignment of authority to make procurement choices, to estimate the costs of switching vendors, and to use the system on a regular basis, cannot be understood without understanding the resolution of bureaucratic conflict in an organization. As shown below, these assignments ultimately determine the influence of switching costs on vendors choice.

b. The tendency to lock-in and bureaucratic conflict

In practice, the principal will have to create a mechanism to induce users to reveal the true costs of switching vendors. If such means cannot be found, the principal has two options. He can either let the users, who have better information about the costs of switching vendors than anyone else, makes those estimates and bias them as expected, or find some other means for making such estimates, such as hiring a technical expert to
resolve the question. The previous discussion suggests that if users choose their vendors and bear the costs of underestimating switching costs, then biases in a user's estimates of the costs of switching to nonincumbents will likely accentuate tendencies to lock-in. If the principal creates a successful monitoring mechanism, then this reduces this tendency to lock-in.

In addition to the relationships already mentioned, the incentives to make far-sighted investments also depends on the outcome of bureaucratic conflict, especially the shape of the funding for the new acquisition. That is, it depends upon whether switching costs are partially or fully covered by a one-time capital budget or whether it is partially covered out of agencies' operating budgets. If switching costs are fully covered in the capital budget, then this induces something like a "moral hazard" problem. Agencies have little incentive to take actions to minimize the cost of a switch if the costs are fully covered, irrespective of the outcome. This should accentuate tendencies to lock-in to incumbents. In the other case, if the agency's operating budget largely covers the costs of switching, their incentives to make investments to limit switching costs are much higher. Due to more programming directed at anticipatory converters, switching costs decline, which may result in more switching. This should decrease tendencies to lock in.

Because the principal will want the agency personnel to take more action to reduce future switching costs than the agency programmers will likely take, there is a clear incentive for the principal to monitor user programming. A principal will want to assess the degree to which an agency is responsible for the level of switching costs realized later. He may also want to counteract the tendency of incumbent vendors to encourage programming practices that raise the costs of switching later (e.g. programming in a system-specific language). The less a programmer internalized those costs, the more likely his actions will accentuate tendencies to lock-in to incumbents. Not surprisingly, such themes can be found throughout the GAO reports written on conversion expenses (Table 5).

A related problem concerns the principal's efforts to monitor and control investment by users and incumbent vendors in relationship-specific assets. The level of effort expended by agencies in vendor-specific assets
will be influenced by the probability that the user will have to switch, especially as the time for a new replacement approaches. Likewise, a vendor has an incentive to invest in factors that raise switching costs, in order to raise the probability that the user will find it too costly to switch to a nonincumbent when an upgrade is required. However, the vendor also has an incentive to neglect those factors that are ignored in the evaluation of new bids. Vendors will not invest in the quality of the relationship between the vendor organization and the government users if it is unaccounted for in the evaluation of bids of incumbents and nonincumbents. Indeed, one recent study shows that all these circumstances can occur (Kelman, 1990).

Agency incentives ultimately depends on the user's discretion level, i.e., whether administrative rules substantially constrain his vendor choice. The totality of the previous analysis argues that if the agency has full discretion over the choice of vendor they tend to lock-in to the incumbent. If the principal tries to change the agent's incentives and closely monitors the choice, the outcome depends on the regulations imposed by the principal on agency decisions. Thus, one open question for this case, and cases similar to it, concerns the principal's ability to counteract the user's bias towards the incumbent with restrictions on agency discretion or with an appropriate contract rewarding verifiably good behavior.

Previous work only partially indicates whether restrictions on agency discretion, usually in the form of administrative rules, reduce the tendency to lock in or not. Work by Marshall, Meuers, and Richards (1990) argues that the rules for protesting vendor decisions largely shape the enforcement of the principal's rules. Under some rules the incentives for losing vendors to protest are greater than is optimal and under others too low. Those protest incentives ultimately shape an agency's incentives to prevent protest, perhaps by estimating switching costs properly. Related work by Greenstein (1991) argues for a slightly different emphasis. That is, the principal is often left with a choice of either applying a procedural rule that is appropriate sometimes, but not too stringent at other times; or letting the inherently biased agent evaluate the inherently subjective information that is relevant to a vendor choice. Different circumstances warrant stringent rules or exceptions to them. When stringent
rules are applied, switching-costs estimates are more likely to be biased downward due to the de-emphasis on intangible future cost. When agencies are given full discretion, the tendency to lock-in is appreciably higher.

The overriding theme of these observations is that the role of switching costs cannot be understood without understanding the assignment of authority for purchasing, switching cost estimation, and system use. All play a role within a more general bureaucratic conflict.44

c. Switching Costs and Organizations in the Private Sector

How relevant are these observations for commercial mainframe use by the private sector? Many of the same fundamental factors observed in federal agencies will continue to influence the role of switching costs in private firms and universities, the two largest users of mainframes. For example, switching costs result from technical incompatibilities and asset durability, which should persist in computer use in private industry; the technical reasons for difficulty in estimation of switching costs will not change in private industry; and the technical links between anticipatory converters and switching costs will not change in private industry; the same principal/agent conflicts and verifiability problems are as likely to arise because system users will continue to informational advantages over their supervisors.45 (See Inmon (1986) for discussion of this problem in the data processing departments of private firms).

What may differ in the private sector is the structure of authority among different individuals within an organization and the coordination mechanisms employed. While it is possible for private organizations to establish performance-based contracts with its computer managers -- e.g., dismiss a manager for poor performance or reward an excellent manager with monetary bonuses -- such instruments are limited by the administrative law governing many public organization, such as federal agencies. In addition, private organizations may not employ a procurement process that follows such distinct and rigid stages as does federal procurement. The relationships between switching costs and particular aspects of a decision will also be less defined within a private firm if administrative rules do not circumscribe an agent's decision-making as much.
More generally, economists tend to think that budget considerations affect the incentives of workers in the public sector less than in the private. The absence of any structural incentives to realize future organization goals might lead to more myopic behavior by federal agencies than one would find in the private sector.

VI. Where to go from here?

With a rare exception, previous economic thinking about switching costs and lock-in has presumed that buyer organizations are monolithic, overlooking the role of decision-making at many levels within an organization. In contrast, this article demonstrated that, for reasons not all of which are peculiar to the federal government, the coordination of decision-making within an organization has important consequences for the role of switching costs. This observation strongly questions the fundamental presumption of previous analysis that buyers can costlessly formulate and implement consistent inter-temporal strategies for dealing with high switching costs. If decisions regarding one component of a system are partially insulated from consideration of decisions related to compatible components, then coordination within organizations will not be likely or even possible.

This conclusion, especially its emphasis on the role of an organization, motivates further research in the economics of market subject to large switching costs. This article argues for different structural assumptions and different questions in future analysis -- e.g. questions that emphasize the coordination of decisions related to vendor choice, the variance of uncertainty regarding the future level of switching costs, and the incentives to invest in anticipatory converters. At the very least, we should expect to find in public and private firms a variety of responses to bureaucratic conflicts. Those responses ultimately determined the observed tendency of buyers to lock-in to incumbent vendors in practice.
Endnotes


3. See International Data Corporation, EDP Newsletter, 12/18/74, 2/12/75, 12/8/75, 1/21/77, 12/5/78, 12/29/80. See Greenstein (1990) for a discussion of these surveys and the difficulties of interpreting them.

4. Indeed, the records used in this article were produced for a variety of reasons related to the topic of technological lock-in.


6. See EDP Industry Reports, various years.

7. See Gray (1981) for a discussion of general purpose applications in the federal government.

8. See the GAO 1980a, Appendix II, and the summary of that appendix. The typical conversion between compatible systems was represented by the IBM 360 and 370, which has low retraining and reconfiguration costs.

9. General purpose mainframe definitions are borrowed from Auerbach reports (1962-1975), Phister (1979), and especially the IDC General Purpose mainframe surveys published in the EDP Industry Reports (1974-1982), and Annual surveys of the industry (1983-1986). This choice excludes all minicomputers, small business computers, desk top systems, and systems sold primarily for dedicated applications, but does include large mainframes for applications which might be called "scientific". The source of definitions guarantees that the systems were widely diffused in the private market as well; hence, the phrase "commercially available." See Greenstein (1989) data appendix for a full definition.

10. This trait is partly due to the sophisticated programmer's tendency to use the most convenient features of a system when writing programs, features that need not be the same on other
machines. One might expect manufacturers to encourage this programming practice as a means to raise switching costs.

11. A limited amount of CPU compatibility across firms did also exist (For example, the RCA 7000 series, IBM360-370 series, Amdahl and National Advanced Systems all roughly fall into the same product family). These constituted a small fraction of total federal sales through the early 1980s. See Greenstein (1990) for an idea of its magnitude.

12. Industry records frequently refer to the incompatibilities of the architecture and system software of the general purpose mainframes produced by IBM, Burroughs, Univac-Sperry, NCR, CDC, Honeywell, DEC, and others (See Auerbach Reports, for example).

13. Competitive procurement means that agencies hold an auction for the right to provide some component of their system. In practice, agencies may anticipate that one vendor is likely to possess a large advantage over all competitors. In this case, the agency many choose to by-pass competitive procedures and sole-source (See Greenstein, 1991).

14. Of course, the average age at replacement is not the same as the expected age at replacement for all processors. In a growing population of processors, it would provide a lower bound. Only in a stationary population would the two coincide. Case studies typically talk about contracting for systems with expected lives of six to eight years. See GAO 1981, for example.

15. There was an analogous phenomena in the typewriter keyboard case. David (1986) notes that the durable asset there was the memorization of keyboards by touch typist. Like software for the human mind, it was costly for some users to reprogram themselves.

16. See Brooks (1971), for a similar emphasis on the technological necessity of solving complex software design problems in sequence, rather than in parallel. Each sub-problem needs to base its approach on solutions to previous problems in the sequence.

17. There is some evidence to suggest that federal users of Control Data systems persisted in holding onto their "orphaned systems," delaying new purchases for a long time (GSA 1987, 1988).

18. The estimated and actual cost (mid-70s dollars) of software conversion alone were large: $1.5 million for software conversion at EPA, 531,000 lines of code converted for an estimated $950,000 at the Navy base in Norfolk, 125,000 lines of applications for an estimated $559,000 at the naval base in Jacksonville, 332 application programs for $486,000 was estimated at the Naval base in Pensacola but 291 programs eventually were converted for $4.5 million, 14 of 571 totally converted and many partially done at a cost of $3.4 million at the USDA in Kansas City, 571 application programs for $3.4 million, 296 programs estimated at $338,000 for the USDA in New Orleans, but which eventually came to several million, and $4.5 million for application software conversion at the VA. In contrast, the one compatible upgrade had software
total conversion expenses of $13,900. These slightly overestimated net conversion costs, because even an upgrade with the same supplier will contain some switching costs, but underestimated switching costs by neglecting some non-pecuniary costs. See Appendix for a conversion costs component breakdown.

19. These are ball park estimates. The first set were computed by taking the very precise estimates of total conversion costs in the case studies and comparing them with the average system price, as listed in the IDC General purpose surveys for that year and 1981 for any earlier conversion (Purchase price estimates did not begin appearing until 1981 and most cases came from the late 70s). They came to 23%, 22%, 27%, 50%, 68%, 79%, 150%, 210%, and 250%. The second set were computed by comparing the same conversion estimates against the IDC average monthly rental for that system in the year of installation. These came to 13, 14, 32, 37, 46, 70, 72, 123, and 128 times the rental price. See appendix.

20. Some allowance in these numbers must be made for the limited technical expertise concerning switching costs in the late 1970s within the federal government. Once switching costs were better understood, the costs of switching should be somewhat lower. However, these studies leave the impression that the costs could only partially be attributed to the limited technical expertise. Even where there was some expertise, as in the military, the costs could still be substantial.

21. This is only commercial systems, or systems for which we can get information about their use in private industry. This excludes many but not all uses that are especially idiosyncratic to the government, commonly found in the Defense and Energy departments. See Greenstein (1989) appendix for a definition of commercial systems.

22. See Heckman and Singer (1982) for example.

23. In addition, Greenstein (1990) argues that evidence is consistent with the view that most switching costs originate with the installation of the first system. In other words, the user pays a one time expense in order to learn how to use a vendor's machines. All subsequent hardware acquisitions after the first do not contribute much to the costs of switching vendors later.

24. In addition, the time horizon for those expectations must be longer in agency use than in private industry. Federal system life is known to exceed private industry system life. See GSA (1987, 1988).

25. In a competitive procurement vendors can bid against one another and have similar products, even if one has a system compatible with the user's, while another does not. If $S$ represents the estimate of the costs converting to a new system, then the level of $S$ largely determines the bidding behavior of the two vendors. These bid prices then determine the likelihood that a user will switch vendors (See Cabral and Greenstein, 1990).
26. Examples of functional loss are numerous. GAO (1980a), reports a case of converting line for line a program that previously took three minutes that then took forty-five minutes to operate on the new system. See GAO (1980a). It also reports a case where a program used to take five hours took twenty-two hours on a new system. It had to be completely rewritten to take advantage of features of the new system (and took only three hours to run when completely rewritten).

27. Strictly speaking, the application need not be a unique one, though all the examples I know typically do involve software that possess some unique features related to the application. Market supplied implementations of software on one system may also not easily transfer to another system. Suppliers of software then might absorb some switching costs if a large number of buyers switch systems. Thus, market supply of software does not eliminate switching costs, though it may spread the incidence of implicit burden between buyers and suppliers of peripheral components. David (1986) makes a similar point during his discussions about the reluctance of typing school instructors and the first touch-typists to memorize alternative (non-QWERTY) keyboards or coordinate their decisions.

28. GAO (1976), p. 20-21 discusses this, often citing programs whose documentation quality was sacrificed for urgent needs of the past, or whose development was done in a patchwork and unsystematic fashion.

29. This last switching cost is typically incurred during "retraining" and does not include nonpecuniary costs such as morale or staff turnover. See GAO (1980a), p. 44.


31. This does not imply, however, that using the most accurate estimate of switching costs is in the interest of the government as a whole. Underestimates induce more switching and also more aggressive bidding from incumbent vendors. The optimal level of switching costs to incorporate will trade off these two factors. See Cabral and Greenstein (1990).

32. See GAO (1977b, 1980a), GSA (1981, 1982, 1983a, 1983b, and 1984) for more information on the management and implementation of conversions and their cost components. Also see GSA (1986), for a well-developed attempt at a conversion cost algorithm, an attempt which demonstrates the inevitable complexity of doing the task in a thorough and complete manner.


34. Despite attempts to the contrary, it appears to be common for even the fairly standardized higher level languages, such as Cobol, to get implemented in incompatible forms on different manufacturer's systems (Bob Dornan, private communication). There are a large number of FIPS publications devoted solely to this subject. See NBS (1977) for a review.
35. See NBS (1977), or any of a large number of FIPS (Federal Information Processing Standards) publications.

36. Not surprisingly, the latter two efforts have had trouble because each agency tended to design and modify programs to its own unique needs, not internalizing whether another agency might want to copy it or desire another modified configuration.

37. It is a perplexing that there exist no extensive discussions in public records of hardware solutions to system incompatibilities, e.g. what are sometimes known as translators. If it were economically viable in some situations, then one would have expected some discussion. Is this silence evidence that these were not viable, or is this silence a function of the sample of problems examined by the writers in the 1970s? The sole exception is one reference to "emulation" - imitation of one system's software by another system's. This discussion does not recommend that emulation be used as a long-term solution to incompatibility problems, citing the inefficient use of hardware resources that results. It was only recommended when an essential database was embedded in an old system where conversion was difficult. See GAO (1980). Of course, there are a large number of attempts to standardize software, as seen in many FIPS publications. Some of this is anticipatory, such as the standardization on ADA and most is not.

38. The following is a short synopsis of the latter half of chapter 2 of my thesis. Similar points can be found in Greenstein (1990).

39. For example, let the incumbent and nonincumbent produce perfect substitutes except for the costs of switching. Let the incumbent have a cost of \( C_I \) and the nonincumbent have a cost of \( C_N \), where the costs of switching to nonincumbents is \( S \). If bidding were characterized by Bertrand behavior, then \( C_I < C_N + S \) implies a winning bid by the incumbent just under \( C_N + S \). But if \( C_I > C_N + S \), then the winning bid is by the nonincumbent and he bids just under \( C_I - S \). The switching cost is then paid by the principal after the switch. In either event, the principal pays for \( S \), either through a higher price or through an actual switch.

40. The present system employs the "technical expert" option. Experts within the Office of Software Development, as sub-agency of GSA, make estimates of the costs of switching vendors and those costs are used in evaluation of competing bids. See OSD (1986).

41. A related issue concerns the rules for accounting for switching costs by the official procurement process. If such costs are ignored, then switching vendors becomes more likely. Yet, the costs of switching decline due to agency investments.

42. See Cabral and Greenstein (1990) for a discussion of how different rules regarding the use of switching costs in vendor choice can yield widely different outcomes.
43. Since government administrative rules only apply to computer procurement of a value greater than $25,000, it is likely that the processes influencing the tendency to lock-in to personal computer, work station, and/or small mini-computer systems is much different from the processes influencing the tendency to lock-in to mainframe hardware.

44. Other aspects about switching costs also change in a bureaucratic setting. It is not always to the advantage of the principal to incorporate fully switching costs into his evaluation of competing bids. If the principal can commit to ignoring switching costs in the evaluation of bids, then such a commitment induces incumbents to bid more aggressively. There are many circumstances in which the extra expenses from occasionally switching vendors on average can be outweighed by the gains from lower bid prices from the incumbent. See Cabral and Greenstein (1990). Indeed, under some circumstances partially accounting for switching costs can be superior to fully accounting for them or to not accounting for them at all. The optimal procurement system, from the principal's perspective, will vary depending on who the incumbents are and the level of information about switching costs held by the vendors, as well as the federal agencies -- trade-offs that economists have yet to fully analyze. Such trade-offs are relevant for federal procurement, where the Congress can commit to bidding evaluation procedures through the use of administrative law and protest procedures. However, such issues may be less relevant for private firms, who may not have any such commitment mechanism available.

45. Due to their technical delicacy, their size, and their complexity, mainframe computers tended to be overseen by specialized departments within private firms. It was not at all unusual in the 1970s and early 1980s for system users and the source of funding to be separate in large private firms.
Table 1.

Number of components in systems in 1979 inventory (Commercially Available General Purpose Systems only)

<table>
<thead>
<tr>
<th>TYPE</th>
<th>SYSTEMS</th>
<th>MEDIAN</th>
<th>MEAN</th>
<th>ST.D.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>2488</td>
<td>1</td>
<td>1.60</td>
<td>3.03</td>
</tr>
<tr>
<td>STORAGE</td>
<td>2488</td>
<td>8</td>
<td>15.8</td>
<td>23.6</td>
</tr>
<tr>
<td>INPUT/OUTPUT</td>
<td>2488</td>
<td>5</td>
<td>14.4</td>
<td>33.9</td>
</tr>
<tr>
<td>TERMINALS</td>
<td>2488</td>
<td>1</td>
<td>14.5</td>
<td>56.6</td>
</tr>
<tr>
<td>OTHER</td>
<td>2488</td>
<td>1</td>
<td>2.64</td>
<td>9.67</td>
</tr>
</tbody>
</table>

Source: ADP inventories, 1979, original data.

- **CPU** stands for any central processing units.
- **Storage units** stands for any of the following: Mag tape, core unit, drum unit, disk unit, misc. storage, multi-purpose control.
- **Input/Output** stand for any of the following: Card reader and/or punch, papertape reader and/or punch, OCR unit, mag data recording unit, mag ink character recognition unit, data converter, media converter, plotter, printer, image handling unit, display unit, operator console, control for IO channels, misc. system IO controls.
- **Communications terminals** stands for any of the following: Card terminal, mag tape terminal, papertape terminal, printer terminal, input console, multiplexor control, misc. terminals and related units.
- **Other** stands for any of the following: EDPE (electronic data processing equipment), card punch, tape punch/verifier, sorter, collator, reproducer/gang punch, interpreter, misc. PCAM or EDPE and unknown.
Table 2.

Number of components in systems in 1979 inventory by manufacturer (Commercially Available General Purpose Systems)

Machine manufacturer same as system designer:

<table>
<thead>
<tr>
<th>TYPE</th>
<th>SYSTEMS</th>
<th>MEDIAN</th>
<th>MEAN</th>
<th>ST.D.</th>
<th>PER</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>2476</td>
<td>1</td>
<td>1.33</td>
<td>1.21</td>
<td>99.5</td>
</tr>
<tr>
<td>STORAGE</td>
<td>2406</td>
<td>6</td>
<td>11.3</td>
<td>16.2</td>
<td>96.7</td>
</tr>
<tr>
<td>INPUT/OUTPUT</td>
<td>2410</td>
<td>5</td>
<td>12.0</td>
<td>27.4</td>
<td>96.8</td>
</tr>
<tr>
<td>TERMINALS</td>
<td>2382</td>
<td>1</td>
<td>10.2</td>
<td>41.6</td>
<td>95.7</td>
</tr>
<tr>
<td>OTHER</td>
<td>2427</td>
<td>1</td>
<td>2.27</td>
<td>8.85</td>
<td>97.5</td>
</tr>
</tbody>
</table>

Machine manufacturer differs from system designer:

<table>
<thead>
<tr>
<th>TYPE</th>
<th>SYSTEMS</th>
<th>MEDIAN</th>
<th>MEAN</th>
<th>ST.D.</th>
<th>PER</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>155</td>
<td>1</td>
<td>4.55</td>
<td>9.72</td>
<td>0.62</td>
</tr>
<tr>
<td>STORAGE</td>
<td>953</td>
<td>2</td>
<td>12.9</td>
<td>23.2</td>
<td>38.3</td>
</tr>
<tr>
<td>INPUT/OUTPUT</td>
<td>746</td>
<td>2</td>
<td>9.23</td>
<td>30.9</td>
<td>30.0</td>
</tr>
<tr>
<td>TERMINALS</td>
<td>1647</td>
<td>1</td>
<td>7.95</td>
<td>44.0</td>
<td>66.2</td>
</tr>
<tr>
<td>OTHER</td>
<td>1962</td>
<td>1</td>
<td>1.47</td>
<td>3.77</td>
<td>78.8</td>
</tr>
</tbody>
</table>

Source: ADP inventories, 1979, original data.

Note: Systems stands for the number of systems with at least one piece of equipment of the designated type and either made by or not made by the system designer.

PER is the percentage of systems with at least one machine of the designated type from the same or different manufacturer out of the total number of systems with any at all.

See Table 1 for remaining definitions.
Table 3

System Supplier for Stock of General Purpose Mainframe Systems.

<table>
<thead>
<tr>
<th>MANU</th>
<th>71</th>
<th>72</th>
<th>73</th>
<th>74</th>
<th>75</th>
<th>76</th>
<th>77</th>
<th>78</th>
<th>79</th>
<th>83</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL</td>
<td>3229</td>
<td>3053</td>
<td>3037</td>
<td>3860</td>
<td>2646</td>
<td>2544</td>
<td>2508</td>
<td>2565</td>
<td>2509</td>
<td>2395</td>
</tr>
</tbody>
</table>


Notes: The table includes only commercially available general purpose mainframe systems, as defined by IDC EDP industry reports (various years), and Digital Equipment Corporation VAX systems. The table only includes acquisition of federal owned or leased systems from external supplier.
Table 4

Commercially Available General Purpose Mainframe Systems acquired each year by Federal agencies from external suppliers

<table>
<thead>
<tr>
<th>Manu</th>
<th>72</th>
<th>73</th>
<th>74</th>
<th>75</th>
<th>76</th>
<th>77</th>
<th>78</th>
<th>79</th>
<th>80-83</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>296</td>
<td>220</td>
<td>279</td>
<td>132</td>
<td>244</td>
<td>97</td>
<td>140</td>
<td>154</td>
<td>720</td>
<td>2282</td>
</tr>
</tbody>
</table>

Notes: Acquisitions were estimated by comparing systems at Federal agency offices in adjacent inventory years. Year is the year the first processor for a system first appeared in the data inventories. Due to unavailability of original data for years 1980, 1981, and 1982, all acquisitions in these years were estimated from inventories for 1983.

The table may overestimate total acquisitions if all intra and inter agency transfers are not recorded, but internal consistency check revealed that this problem is not likely to be large.
Table 5

Oversight Reports Concerned with Conversion Problems: 1977 - 1986

General Accounting Office Publications:


General Services Administration Publications:

Office of Software Development and Information Technology:


1983, Software Conversion Lessons Learned, Volume I, FCSC-83/003, 1983


National Bureau of Standards Publications:


REFERENCES


Farrell, Joe, "Competition with Lock-in," University of California working paper, January 1987


------, "Installed Base and Compatibility," AER, 1986


International Data Corporation, EDP Industry Report, Framingham, Massachusetts. Various years.


**U.S. Government Publications:**

*General Accounting Office Publications:*


1977b, "Millions in Savings Possible in Converting Programs from one Computer to Another," Sept. 15, 1977, FGMSD-77-34.


General Services Administration Publications:


Office of Software Development and Information Technology:


1983b, Software Conversion Lessons Learned, Volume I, FCSC-83/003, 1983


National Bureau of Standards Publications:


