Price Indexes for PC Database Software and the Value of Code Compatibility

by

Dietmar Harhoff* and Dietmar Moch**

** Zentrum für Europäische Wirtschaftsforschung (ZEW),*
University of Mannheim and CEPR
*** Zentrum für Europäische Wirtschaftsforschung (ZEW)*

December 1996

Keywords: price indexes, hedonic methods, technical change

Abstract

Changing product quality poses a challenge for the computation of price indexes, in particular in technologically advanced industries. We assess the differences between traditional and quality-corrected indexes by computing hedonic and matched-model price indexes for personal computer database software. Our database covers the price development in Germany from 1986 to 1994. Quality-adjusted software prices *decline* by 7.4 percent according to our hedonic index. Surprisingly, a matched-model index based on linking the prices of directly comparable program versions decreases even faster than the hedonic index (9.3 percent). This unusual result is apparently caused by the simultaneous selling of old and new versions of a given software product. The estimation results also confirm the importance of network effects. Code compatibility, i.e. the capability of executing programs written for the dominant database product, yields a significant price premium. The ability to read and write data in the dominant spreadsheet format (file compatibility) is also associated with higher prices, but the price differential is much smaller than in the

case of code compatibility.

Non-Technical Summary

The computation of price indexes is often considered a rather mundane and boring task in economics. Yet, given the importance that is attached to the use of price indexes in economic policy decision-making, it is obviously an important one. In the context of goods with changing product quality, it can often be a complex task, too, since it is difficult to seperate quality changes from the pure time component in price series. To study the implications of various methods used in the computation of price indexes, this paper presents empirical evidence for a rather narrowly defined group of software products: PC-based database software. We consider several types of indexes in this paper and explore how the results are affected by changes in the computations.

Our database covers the price development in Germany from 1986 to 1994. Quality-adjusted software prices *decline* by 7.4 percent according to our hedonic index. Surprisingly, a matched-model index based on linking the prices of directly comparable program versions decreases even faster than the hedonic index (9.3 percent). This is an unusual result, since most previous studies find that matched-model indexes typically understate price declines (or, respectively, overstate price increases). The result is no artefact, but apparently caused by the simultaneous selling of old and new versions of a given software product. The estimation results also confirm the importance of network effects for PC database software products. Code compatibility, i.e the compability of executing programs written for the dominant database product, yields a significant price premium. The ability to read and write data in the dominant spreadsheet format (file compatibility) is also associated with higher prices, but the price differential is much smaller than in the case of code compatibility. This effect is consistent with theoretical arguments: changing dedicated software to a new standard should be considerably more costly than simply transforming a data format.

1 Introduction

Information technology has become an important segment of production and services. For example, the German information technology (IT) market was valued at about 37 Billion ECU in 1994 - equivalent to 4.5 percent of Germany's Gross Domestic Product. Most of these goods and services are demanded by firms, but the household share of IT consumption has been increasing lately. During the last decades, product and service quality has changed dramatically in the information technology market, particularly in computer hardware and software. At the same time, nominal prices have either decreased slightly or just maintained their level. It is well-known by now that in the presence of substantial changes in product and service quality, standard methods of computing price indexes may lead to biased results (Griliches 1971). Such biases can have a number of drastic consequences and may be particularly relevant for the formulation of research policies. For example, studies of productivity growth depend crucially on the availability of correct price index series. The impact of IT on productivity growth has caught the attention of many researchers and policy-makers, but any results obtained on the basis of questionable price indexes (either on the input or output side) will not provide conclusive evidence of the impact of, say, the application of modern computer hardware and software.[1] Moreover, the results of international comparisons are likely to be distorted by the use of price indexes originating from different methods of calculation.[2] With a growing proportion of *household consumption* being devoted to IT goods and new information services, one may also wonder whether current inflation estimates correctly measure overall consumer price trends. The recent debate in the US concerning the validity of the CPI (consumer price index) underlines the necessity for a detailed assessment of existing techniques used in the computation of price indexes.[3] If the increasing quality of IT goods and services is not sufficiently captured in official inflation estimates, this might lead to over- or underestimation of inflation. Since many macroeconomic policies are contingent on the rate of inflation, errors in estimating the inflation rate

---

1 For a detailed discussion and new research results see Brynjolfsson and Hitt (1996).

2 For example, Mairesse and Hall (1994) argue that their comparison of R&D effects in France and the U.S. is affected by this problem, since hedonic indexes have been introduced in the US statistical system while they are not yet being used in France.

3 See Moulton (1996) for a summary and discussion of the CPI debate.

may directly lead to erroneous decisions in economic policy. While this statement holds in general, potentially erroneous conclusions regarding the impact of R&D or of information technology are likely to be most pronounced in technologically advanced industries. Thus, the formulation of R&D policies is likely to be affected by problems of price index measurement.

The computation of hedonic price indexes which control for quality changes has received considerable attention as a potential safeguard against such distortions. Hardware components in the computer industry have been among the favorite examples for the application of hedonic price indexes. Of particular relevance for this article are a number of recent studies on the decline of quality-controlled prices for personal computers (PCs). Gordon (1990), Berndt and Griliches (1993), Nelson, Tanguay, and Patterson (1994) and Berndt, Griliches and Rappaport (1995) have shown that quality-adjusted prices for PCs decline on average by 25-30 percent per annum. The evidence for computer software is less well-developed. A study by Gandal (1994) yields a yearly average decline of 15 percent for list prices of spreadsheets.4 A more recent paper by Gandal (1995) studies the determinants of prices for database software, with particular attention being paid to the effect of compatibility standards, but does not develop detailed data on quality adjusted prices. This paper is meant to add to the existing literature by providing some new evidence on the development of software prices. Using a relatively large sample of observations on prices and product characteristics, we compute various price indexes for PC software in the Federal Republic of Germany. In order to focus on a reasonably homogeneous software market, we analyze the price development of standard *database* software products for MS-DOS compatible Personal Computers. Together with word-processing software and spreadsheets, database products belong to the commercially most important software products. According to estimates by the International Data Corporation (IDC), more than 750,000 licenses for this group of products were sold in 1994 in Germany. According to IDC forecasts 2.5 million packages will be sold in 1998. Using an average price estimate from our data of about DM 1000 per package in 1994, database software alone would account for 1994 revenues of about DM 750 million, approximately 4 percent of the total computer hardware and software market.

---

4 Gandal’s (1994, 1995) articles appear to be the only empirical studies of PC software prices to date.

Our empirical results demonstrate the importance of quality changes for the price development in the PC database software market. Matched model indexes generally tend to underestimate the price decline, but we show that the level at which the match is constructed turns out to be quite important. On average, prices decline by 7.4 percent according to our hedonic index. Surprisingly, the matched-model index based on linking the prices of directly comparable program *versions* decreases even faster than the hedonic index (9.3 percent). This result is apparently driven by a particular feature of the software market. Whenever a new version of a software package appears, the previous version does not disappear immediately from the market, but its price is driven down due to the arrival of the improved software. In matched model price indexes, the new version is not included in the year of its introduction, but the reduced prices of the older version contribute to an excessive decline in the price index computed from matched versions.

The estimation results also confirm the importance of network effects. Contrary to previous studies, we emphasize a crucial distinction between file compatibility and code compatibility. The former type of compatibility allows users to share their data, while the latter allows them to share programs. Since the latter form of compatibility protects past investments in programming and human capital, we expect that code compatibility is more valuable for software users. Consistent with our hypothesis, we find that *code compatibility* with the dominant database software package *dBASE* yields a significant price premium. Moreover, the ability to read and write data in *dBASE*-compatible database format and *LOTUS*-compatible spreadsheet formats is also associated with higher prices, but the price differential is considerably smaller than in the case of code compatibility. These results differ considerably from empirical estimates obtained in previous studies (e.g., Gandal 1995).

The paper proceeds as follows. In section II, we briefly describe the theoretical background of hedonic price estimation techniques. In section III, we introduce the dataset and the variables used to model the quality of PC database products. In section IV, we present regression results for hedonic indexes and the computation of a number of matched-model price indexes. We compare these estimates and comment on the differences between them. Section V summarizes the empirical results from this study and draws some conclusions for future work.

2 Theoretical Background

Since the problem of obtaining price indexes in the presence of quality changes has received considerable attention lately, we only provide a brief summary of the theoretical background. Detailed discussions have been given, *inter alia*, by Griliches (1971), Muellbauer (1974) and Rosen (1974). A critical discussion of the hedonic method and the description of alternatives is provided by Trajtenberg (1990).

Statistical agencies often use the matched-model method to compute price series. The resulting indexes are also referred to as chained price indexes. This procedure is described in detail by Triplett (1990) for the United States and in Szenzenstein (1995) for Germany. In essence, the price index is computed as the ratio of successive prices for the same good (or bundle of goods). Obviously, in employing the matched model approach a decision has to be made whether a given product is similar enough to some product from the previous period such that a link can be established. The price index is then computed as a weighted average over all price changes computed from linked products. This classical method is appropriate for products that maintain their quality charateristics over time either fully or to a large extent. Thus, computing price indexes for commodities like crude oil, cement, or grain, to give just a few examples, is indeed a relatively simple task.

In many cases, however, the chaining of products may either be infeasible (e.g., if a completely new product appears) or highly unreliable, e.g., if the products whose prices are chained intertemporarily are subject to considerable quality changes. Thus it is considerably more difficult to obtain price indexes for goods that can be thought of as bundles of characteristics which are subject to change. Personal computers are a particularly suitable example, since it is intuitively clear that a direct comparison of a typical PC bought at the beginning of the Eighties bears little resemblance with a typical personal computer of more recent vintage. In objective terms, such as computational speed and storage capacity, the quality of this product has increased considerably. Suppose for a moment that the nominal prices of computers had remained at the same level. A naive chained price index which does not take quality changes into account would show no signs of either increase or decline. Yet, given that quality has improved, it is clear that the true price index should be declining.

The hedonic technique attempts to decompose the price of a good into a quality and a time component. Only the latter component is taken into account when the price index is computed. To be more specific, assume that the structural part of a hedonic price equation is given as $g(p)=f(X,\beta)$. Here, p is the observed price of the product, X represents a vector of variables describing the product's quality characteristics, and β is a vector of parameters to be estimated. The functions $g(.)$ und $f(.)$ are often assumed to be linear or log-linear in their respective arguments. Like most other studies, we will use a double-logarithmic specification of the type $\log p = \log X \beta$ for the structural part of the equation such that the parameter vector β can be interpreted as a vector of elasticities.

The computation of the price index from this specification can best be explained in a simple example. Assume that there are price and quality observations for a large number of products for three years (say 1990, 1991, and 1992). For simplicity let us assume further that only one quality characteristic x is of relevance. The relevant price-quality regression equation for product i in year t is then given by

$$\ln p_{it} = \alpha + \beta \ln x_{it} + \delta_{91} D_{91} + \delta_{92} D_{92} + \varepsilon_{it}$$

where the disturbance terms are assumed to be independently distributed5 and the year 1990 is taken to be the reference year for the time dummy variables D_{91} and D_{92}. In this case, the price index can be computed in a particularly simple way. Holding product quality characteristics x_{it} constant, the parameter estimate $\hat{\delta}_{91}$ is identical to the estimated price difference $\ln \hat{p}_{91} - \ln \hat{p}_{90}$. The estimated price ratio and thus the price index is then given by $\exp(\hat{\delta}_{91})$. The exponentiated coefficient of the time dummy variables is therefore an estimate of the respective price index relative to the reference year. This is the approach taken in the following sections. Before turning to the actual estimation exercise, we discuss data collection routines and the relevant quality characteristics of PC database products.

3 Data on PC Database Products and Variable Definitions

3.1 Data Sources and Collection

The empirical use of hedonic techniques to construct price indexes requires detailed information on prices *and* quality-related product characteristics. To study the relationship between quality and product price, we collected a relatively large dataset on PC database software products that were sold from 1986 to 1994 in the

5 In our empirical study, we use the heteroskedasticity-robust variance-covariance proposed by White (1980) to account for non-identically distributed error terms.

Federal Republic of Germany. Only products that included German language support and a German language manual were selected. This restriction excludes a number of products like *Quicksilver* or *dbXL* that were imported from the US, but only available in an English language version. These products were of some importance in the US market, but never gained commercial relevance in Germany. Moreover, we confined our efforts to construct price-indexes for PC database software compatible with MS-DOS. Products for the *Macintosh* and other computers not compatible with MS-DOS were not included. Furthermore, we excluded software packages that were exclusively compatible with *MS-WINDOWS* or *OS/2* operating systems. All of the packages taken into account are stand-alone applications, i.e. they are not bundled with spreadsheet or word-processing products as has been the case with a number of business applications. Finally, public domain and shareware programs were excluded because - contrary to the US - they did not become an important segment of the German database software market.

For a number of reasons, we chose to sample prices from mail order advertisements. First, a price index should record real transaction prices as closely as possible. Arguably, mail order prices are better measures of actual transactions prices than - say - list prices which are often subject to considerable rebates. The mail order market was also chosen because it is the economically most important distribution channel for PC software in Germany. Price data were collected retrospectively from advertisements in *CHIP* which until 1992 used to be the leading magazine for personal computing in Germany. This magazine contains advertisements of retailers that offer their products to private buyers and to businesses alike. In the years 1993 and 1994, many software retailers shifted their ads from *CHIP* to *PC-Direkt*, another magazine for PC users. Hence, the *CHIP* advertisements in 1993 and 1994 were complemented by ads in *PC-Direkt*. Advertisements from the same retailer appearing in both of these magazines were recorded only once.

The data include observations from the years 1986 to 1994 with prices taken from the April, August and December issues of the PC magazines of the respective year.6 We collect the prices every four months to cover the full information about price development, since we expect a sharp price decline. If only yearly information was

6 Note that the magazine issue for a given month is actually available about 2 to 3 weeks prior to the beginning of the month. The April issue typically contains prices for new products that are being introduced at the yearly COMDEX and CeBit trade fairs. The December issue is chosen, since mail orders are particularly high in that month.

used the results might depend on the point of time when the data are collected. Because data on market shares are not available, we used the frequency of product advertising as a rough approximation for the product's market share. All advertisements that were as large as or larger than one quarter of a standard page in these magazines were included in the data collection. Advertisements in *PC-Direkt* had to meet an additional condition: only retailers that had advertisements in earlier issues of CHIP were considered in order to generate comparable data. Using all selected advertisements from the three respective issues of *CHIP* and *PC-Direkt* between 1986 and 1994, we obtained a sample of 711 price observations for 67 distinct versions of PC database software. The composition of the sample reflects the growing importance of PC database software: for 1986, we obtained only 23 observations, but the number of observed software prices increases steadily to 143 in 1991 and then declines until 1994 (47 software price observations) due to the increasing popularity of WINDOWS-compatible products which are not included in our sample.7

3.2 Relevant Quality Characteristics and Definitions of Variables

Hedonic regressions require information on the product's quality characteristics as independent variables. Typically, product quality is difficult to observe retrospectively, since information on previous versions of a software product is hard to obtain. For the purpose of this study, the relevant quality information had to be combined from various sources. Several computer magazines8 were consulted for tests and product specifications. Manuals and documentation were collected and searched for information. In some cases, we simply obtained an old version of the respective product and generated the required data via experimentation.9 In our hedonic regressions, we concentrate on four dimensions of quality and attempt to measure the software product's quality using objectively measurable characteristics of the product.

7 Our sample of WINDOWS compatible products was too small to be included in this study. However, inspection of the trade press indicated that this new standard also affected the relevant quality characteristics of the database products. Thus, a joint treatment in one regression equation may not be possible without introducing numerous interaction terms.

8 The magazines were CHIP, c't, PC Direkt and PC Professional.

9 See Moch (1995) for more information about data collection.

Capacity. In early versions of database software packages, the maximum number of fields or the maximum number of records in a database turned out to be effective constraints for users. These limitations were relaxed during the late 80s, as the number of records that could be managed rose to one billion. The variable *LNFILES* is defined the natural logarithm of number of files that can be opened simultaneously. It represents a measure of capacity, since there is a high correlation between number of files, number of indexes, number of fields per table and the data-types that are available.

Compatibility. Adherence to standards and compatibility is a very important quality feature for software products (Gandal 1994, 1995). But while compatibility has often been interpreted as a one-dimensional concept, it is worthwhile to distinguish between two levels of compatibility. The first level (*file compatibility*) allows the product to directly read and write formats used by other software products without requiring time-consuming data transfer processes. The second level of compatibility is *code compatibility* and allows the sharing of program code, thereby making it easier to switch between various programming environments. The difference between code compatibility and file compatibility may be important. In the case of database software, a package may be able to read data stored in *dBASE* format, but may not be able to run a program coded in the *dBASE* programming language.10 Typically, this is the case for low-cost database software products. Note that file compatibility may assure users that their data can be used with other packages, while code compatibility also protects past investments in programming. We would therefore expect that the latter mode of compatibility is even more valuable to users than the former.

We use three variables to describe the degree of compatibility. Code compatibility is described by the *XBASE* dummy which takes the value 1 if the product can process program code in *dBASE* language, and otherwise zero. File compatibility with the dominant spreadsheet format is captured by the *LOTUS* dummy which takes the value 1 if import and export of data in *LOTUS* format is possible and zero

10 Examples for packages that may read dBASE files, but cannot execute dBASE programs include DAVID and RapidFile. As an example for lack of either type of compatibility, no version of the PARADOX package is capable of executing dBASE programs, and none of the PARADOX versions except for version 4.5 can read data in dBASE format.

otherwise.11 File compatibility with the *dBASE* format is summarized in the dummy variable *DBF* which takes the value 1 if the package can read and write data in the *dBASE* file format and zero otherwise.

Network support. If users want to share data in multi-user environments, database software has to support concurrent requests. A function that is particularly useful to manage concurrent requests is the record-locking feature. The *RECLOCK* dummy takes the value 1 if record locking is possible, and zero otherwise.

Speed. There are numerous criteria describing the performance of database software, e.g., time needed to seek, to sort or to reindex data. The benchmark results differ widely and depend heavily on the hardware the tests are run on. There are no performance indexes offering a consistent testing scheme for all products in our dataset. We therefore use the dummy variable *COMPILER* as an approximation for the product's performance, since compilation of programs will usually result in greatly enhanced speed of database applications. This dummy takes the value 1 if a code compiler generating high speed applications is available, and zero otherwise.

Our data also allow us to explore the relevance of user expectations with respect to the future availability of a given product. Since a user's investment may be at risk if updated versions of his particular software are not available in the future, uncertainty of this type will be equivalent to the perception of lower product quality. This is even more likely to be the case if the product used is not compatible to other software packages. While our data are not experimental, the development of the software market for database products generated one interesting quasi-experiment in this regard. Up to 1991, *Borland* had offered only one database product, the *PARADOX* package which offered a number of unusual features and high product quality, but low compatibility to other products. In particular, the package was not code compatible to *dBASE* programs. In summer 1991, *Borland* acquired *Ashton-Tate*, the producer of *dBASE*. That step cast considerable doubt on the future prospects that *PARADOX* would be continued as a software platform. Since these expectations may have had dramatic effects on the perceived quality of

---

11 In our sample, there is no package that cannot import ASCII code directly. For other file formats like SYLK and DIF we did not have complete information for all of the software packages in our sample, but we did not find any significant effects in a sample of 367 observations for which the respective information was available. This result is consistent with estimates presented by Gandal (1995) who does not find any significant price premium for file compatibility file formats other than LOTUS.

the *PARADOX* software, we use two dummy variables to capture these changes. The *PARADOX86_90* dummy takes the value 1 if a *PARADOX* product is observed in the years 1986 to 1990, and zero otherwise. The *PARADOX91_94* dummy takes the value 1 if a *PARADOX* product is observed in the years 1991 to 1994, and zero otherwise. Since the product had a number of unusual features which were highly valued by some market segments (e.g. professional programmers), the two dummies may capture quality characteristics that are not correlated with the other independent variables12. Of particular interest is therefore the difference between the respective coefficients.

Gandal (1994, 1995) uses yearly list prices per product in his hedonic regressions.13 Contrary to his study, we have available a rather large dataset of actual transaction prices which allow us to estimate hedonic prices directly from micro-data. As our dependent variable we use the natural logarithm of the recorded product price as it is announced in the advertisement (*LNPRICE*). Customers have to pay an additional shipping charge of about DM 20 which we did not include in our price measure. The price does include a value-added tax of 14 percent from 1986 to 1993, and of 15 percent since then. We adjusted price observations from 1986 to 1993 to be comparable with the higher VAT level of the later period. Since prices enter our specification in logarithmic form, the presence of a time-invariant multiplier due to the VAT does not affect our results.

To describe the time trend of prices, we use yearly dummies *YEAR87, ..., YEAR94* that take the value 1 if the observation is in the corresponding year and zero otherwise. We also introduce dummy variables which allow us to analyze the price trend within a given year. The *APRIL* dummy takes the value 1 if the observation was taken from an April issue of the PC magazine, and zero otherwise. The *AUGUST* dummy takes the value 1 if the observation was taken from an August issue, and zero otherwise. 1986 is used as the reference year and December is the reference month in our regressions. Thus, both the *APRIL* and the *AUGUST* dummy

---

12 Since these features were not included in any other package, we do not code them separately. They would be perfectly collinear with a dummy variable for PARADOX.

13 Gandal's (1995) study is based on a much shorter time period (1989-1991) than our study which analyzes the development of prices for PC database software over the period from 1986 to 1994. The relative lack of longitudinal information on prices may explain why there does not appear to be a systematic time trend of software prices in Gandal's (1995) results.

variable coefficients should be positive if we assume that price decline during the respective year.

4 Price Indexes

4.1 Hedonic Price Index

Three sets of regression results are summarized in Table 1. In column (1), we simply regress the price variable on time dummies. The dummy variables then reflect the decline of *average* prices in our sample. The results show that by 1994, average prices had declined by about 35 percent from their initial 1986 levels. While we can compute a price index on the basis of these dummies, that index would be subject to the criticism that quality improvements are not taken into account. It will therefore merely serve as a baseline result against which our hedonic estimates computed from the results in columns (2) and (3) can be compared.

In column (2), we use the full micro-data sample in order to estimate the price-quality relationship. Our theoretical expectations are broadly confirmed in these estimates. Greater capacity (higher values of *LNFILES*) are associated with significantly higher prices for database products, the respective elasticity being about 0.082 (standard error 0.015). Moreover, network effects are clearly visible in our estimates: products that offer compatibility with the dominant spreadsheet format (*LOTUS*) yield prices that are 18.5 percent higher than for incompatible software. This is consistent with estimates by Gandal (1995), but the *LOTUS* coefficient in our regressions is only half the size of Gandal’s. Code compatibility with the *dBASE* programming language (*XBASE*) yields prices that are significantly higher than for software packages lacking this feature.

Table 1

Hedonic Regression Results

(Heteroskedasticity-robust standard errors in parentheses)

Independent Variable	Mean (S.E.)	Specification (1)	(2)	(3)
LNFILES	3.624 (1.474)		.082 (.015)	.076 (.015)
LOTUS	.529		.171 (.048)	.147 (.049)
DBF	.821			.251 (.076)
XBASE	.799		.750 (.108)	.775 (.109)
RECLOCK	.734		.087 (.043)	.106 (.046)
COMPILER	.696		.088 (.045)	.057 (.049)
PARADOX86_90	.098		.957 (.107)	1.195 (.129)
PARADOX91_94	.086		.149 (.121)	.385 (.131)
YEAR87	.052	-.034 (.064)	-.038 (.046)	-.042 (.048)
YEAR88	.098	-.176 (.053)	-.199 (.043)	-.203 (.045)
YEAR89	.156	-.133 (.050)	-.185 (.040)	-.186 (.042)
YEAR90	.198	-.179 (.051)	-.245 (.040)	-.249 (.041)
YEAR91	.201	-.359 (.057)	-.319 (.041)	-.317 (.043)
YEAR92	.114	-.336 (.055)	-.359 (.040)	-.355 (.042)
YEAR93	.093	-.353 (.059)	-.435 (.045)	-.433 (.047)
YEAR94	.052	-.436 (.072)	-.621 (.070)	-.633 (.071)
APRIL	.345	.050 (.028)	.067 (.018)	.068 (.018)
AUGUST	.312	.044 (.027)	.031 (.019)	.032 .(019)
CONSTANT		7.410 (.047)	6.227 (.121)	5.999 (.130)
Observations		711	711	711
R-squared		.134	.623	.625
Adj R-squared		.121	.613	.615
S.E.E.		.305	.202	.201

Note: The dependent variable is LNPRICE, the logarithm of the software package's advertised price. Reference year is 1986, reference month is December. Heteroskedasticity-robust standard errors were computed using White's (1980) variance-covariance estimator.

The estimated price differential is surprisingly large (about 110 percent), but closer inspection of our data reveals that low-cost database packages are typically lacking the feature of code compatibility with the *dBASE* programming language.

In column (3), we also include a dummy variable for simple file compatibility with the *DBF* format as an additional regressor. According to our results, prices for *DBF* compatible products are about 28 percent higher than for packages that are not file-compatible while the price differential for code compatibility with the dBASE language remains large and statistically significant. Thus, the presumption that code compatibility is more valuable than file compatibility appears justified. Code compatibility is technically much more difficult to achieve than file compatibility, and the economic benefit to users of code compatible packages can be substantial, since their past investments in terms of programming own applications are safeguarded by using code compatible software.

Gandal (1995) does not report any significant effect of compatibility with the *dBASE* file format, but his study only refers to the software's ability to read and write in a particular data format (file compatibility), while we also consider the software's ability to execute *dBASE* programs (code compatibility). The differences between our and Gandal's results may arise since file compatibility is easy to achieve such that low-cost products are usually file-compatible with *dBASE*, but not code-compatible. Interestingly, when we exclude the *XBASE* dummy variable for code compatibility from our regressions, we obtain estimates that are remarkably similar to Gandal's: the estimated coefficient (standard error) for file compatibility with *dBASE* is 0.045 (0.093) and the *LOTUS* dummy assumes the value of 0.301 (0.059) in this case. Gandal obtains coefficients (standard errors) of -0.13 (0.12) and 0.31 (0.14), respectively. It appears that the definition of compatibility matters considerably, and that not taking account of code compatibility leads to an important omitted variables bias.

Both record-locking (*RECLOCK*) which is a major element of network operability, and the ability to produce compiled and subsequently fast applications (*COMPILER*) are positively correlated with the package's price, although the coefficients are not as highly significant as the ones discussed before. These coefficients follow our expectations since improvements in quality should be associated with higher prices. Finally, Borland's acquisition of *Ashton-Tate* in 1991 apparently had some effect on software prices. The price for the *PARADOX* package was considerably lower in the period after the take-over of Ashton-Tate. This may reflect the fact that the *PARADOX* package was perceived to be of lower benefit to users after uncertainty with regard to its future availability had become wide-spread. Note, however, that the ex ante price was considerably above-average, presumably due to the large number of extraordinary features embodied in this product.

One should note that the price decline measured by the time dummy variables in columns (2) and (3) is stronger than in the first regression, i.e. one would underestimate the price decline if one were willing to rely simply on an analysis of average prices without taking quality changes into account. In our estimates in column (3), the overall decline is 46 percent, 11 percentage points higher than in the case of the regression in column (1). The estimates presented in Table 1 may be sensitive to our weighting scheme which is based on the frequency of advertising. To test for such problems, we also estimated the hedonic equations using a reduced sample of 67 average yearly prices for each of the software versions in the original sample. The results are quite robust to this change. The coefficients of the quality variables do not change in any major way (compared to the standard deviations of the coeffiecients), and the resulting price indexes do not differ considerably from the one computed from the full sample. The results from the reduced sample thus support the notion that our hedonic estimates are quite robust. Moreover, we obtain results that are very similar to those in column (3) of Table 1 if we exclude all observations from 1992 to 1994. Thus, our estimates are apparently not affected by the increasing use of *WINDOWS* compatible products in those years.

4.2 Matched-Model Indexes

To test for differences between the hedonic index and indexes based on the matched-model procedure, we apply the latter at three different levels of aggregation. First, we establish links at the level of software versions, i.e. we deem products comparable only if the version of the software remains the same. At the product level, we disregard differences between versions, but we consider *dBASEIII*, *dBASEIV* and *dBASEV* as distinct products. Finally, disregarding different products, we only compute prices at the brand level which assures that there are virtually no missing price data due to new products, except in cases when a new brand is introduced or another brand exits the market. Again, we use the number of observations at any particular level of aggregation to generate a weighting scheme, since the advertising behavior of the producers is likely to reflect their relative market position.14

The results from these computations and the hedonic indexes are summarized in Table 2 together with the average annual price change computed from each of the price series. Typically, matched-model indexes yield smaller price declines than hedonic indexes, although theoretically, both over- or underestimation may occur. In our case, the level of aggregation for the matched-model index appears to exert a strong influence on the results. The hedonic index declines on average by about 7.41 percent per year, while the corresponding change based on average prices or matched-model procedures at the level of brands and products yield smaller estimates. Apparently, the higher the level of aggregation the stronger is the tendency to underestimate the decline of software prices. Standard methods of computing price indexes for products whose quality characteristics are subject to considerable change may theoretically over- or underestimate the true (i.e. quality-adjusted) price development (Triplett 1971, Burstein 1961). However, most empirical studies have found that the hedonic index tends to indicate a smaller extent of price increases, respectively a stronger decline of prices over time. It is somewhat surprising therefore that the price index based on matching software versions yields an average annual decline of prices of 9.25 percent which is higher than the one computed from the hedonic estimates. Closer inspection of the data reveals an important phenomenon which is specific to the software market. Apparently, when a new version is introduced, the preceding one is still made available to buyers, but at sharply reduced prices. This reduced price will still enter the computation of a matched-model price index while the presumably higher price for the new version is not included, since no comparable version to which the new one can be matched existed in the previous period. For example, for 1992, the year of the introduction of *dBASEIV* Version 1.5, no link for this software version can be established, since this version has not been in existence before. Yet, *dBASEIII* Version 1.1 was still available in that year, and its average price fell from DM 1481 in 1991 to DM 935 in 1992 due to the introduction of the improved product

---

14 Weights are computed as follows. If a product available in period t is matched to a product available in period t+1, then the respective price index is computed as the ratio of average prices in the two periods. In computing the overall price index as a weighted sum of the individual indexes, this particular ratio is weighted by the sum of the number of observations of the respective product in periods t and t+1 divided by the total number of linked observations.

dBASEIV Version 1.5. This effect does not play a major role once matching occurs at the level of products or brands since the new appearance of products or brands is relatively infrequent. Conversely, new versions appear about every two years and replace older ones.

For the same reason, matching on the basis of major releases (products) ignores the pricing of the newcomers *PARADOX* and *Foxbase* in 1987. Thus, the respective index even shows an increase in these years due to increasing prices of *dBASE* and *Clipper*, while the newcomers entered the market at a lower price. Average and hedonic price indexes include the pricing of the newcomer products and decline in this year. Again the matched model index is distorted because it neither encompasses the completely new products nor does it reflect quality improvements in existing ones correctly.

The slight increase in 1989 of all indexes except for the version-based matched model index can be explained by the introduction of a new generation including *dBASEIV* and *Foxpro*. The quality-adjusted index reflects the improved quality and shows only a minimal price increase. The version-based matching index does not consider the new, expensive versions and therefore the index does not increase. The tighter the matching rules for linked indexes are, the more sensitive the index will be to different levels of quality. But simultaneously, the problem of missing prices becomes more serious, since the respective prices only enter in the second year after the new version has been introduced. The hedonic method corrects for both problems as long as the products have common characteristics and quality change can be described on the basis of observable characteristics.

Table 2

Price Indexes for PC Database Software 1986-1994

(1986=1.00)

	Year									
Price Index	1986	1987	1988	1989	1990	1991	1992	1993	1994	AAGR
Matched Model Index (Version)	1.00	1.05	0.79	0.77	0.75	0.61	0.56	0.52	0.46	- 9.25
Matched Model Index (Product)	1.00	1.04	0.94	1.05	1.09	0.87	0.79	0.74	0.70	- 4.36

Matched Model Index (Brand)	1.00	1.04	0.92	0.99	0.98	0.84	0.82	0.80	0.73	- 3.86
Index based on Average Prices	1.00	0.97	0.83	0.88	0.83	0.70	0.71	0.71	0.65	- 5.24
Hedonic Price Index	1.00	0.96	0.82	0.83	0.78	0.73	0.70	0.65	0.54	- 7.41

Note: AAGR is the average annual growth rate.

5 Conclusions

The results of this study demonstrate that quality-adjusted prices for PC database-software in Germany have declined by about 7.4 percent during the past nine years. But the average annual price decline is only about one third of the average annual rate that has been observed for PC hardware in the United States. Similarly, another study of software prices (Gandal 1994) arrives at an estimate for the average price decline of 14 percent per annum in the case of spreadsheet packages. While there may also be differences in the market structures between the US and Germany there are also product specific differences. The market for spreadsheets has grown faster and earlier than the market for database software. As software production has high fixed cost for development and low variable production cost, one would expect stronger price decline in larger markets. The difference between the findings of Gandal and our results may also be explained by the fact that Gandal uses list prices while we are using transaction prices. In Germany and in the U.S., software producers were forced to bring down list prices because list prices were about two or three times higher than the respective „street“ prices. Assuming that the companies' policy on list prices has been a global one, we would expect a sharper price decline in list prices than in „street“ prices.

These results appear to suggest that software prices decline at a slower rate than prices for PC hardware. Our estimates cannot reveal whether this outcome is due to extensive market power in the software market, but they could be consistent with this hypothesis. Market power of software producers may also be stronger in Germany due to a lack of fringe products offered by smaller U.S. producers which do not export their products. However, strenghtening these arguments would require a structural framework for the analysis of software producer interaction.

Our study also points to significant benefits from code compatibility. The capability of executing programs that are written in the industry's standard programming language (*dBASE*) results in a large and highly significant price differential. File compatibility with the dominant standard format for spreadsheets also yields a significant, but much smaller differential. To our best knowledge, this distinction between code and file compatibility has not been made in prior empirical studies of software prices, but it appears to be of some relevance.

6 References

Berndt, E. R. and Z. Griliches (1993). "Price index for microcomputers: An exploratory study," in: M. Foss, M. Manser and A. Young (eds.): Price measurements and their uses. Studies in income and wealth, Vol. 57, pp. 63-93. Chicago: Unversity of Chicago Press.

Berndt, E. R., Z. Griliches and N. Rappaport (1995). "Econometric estimates of price Indexes for personal computers in the 1990s," Journal of Econometrics, Vol. 68, pp. 243-268.

Brynjolfsson, E. and L. Hitt (1996). „Paradox lost? Firm-level evidence on the returns to information systems spending," Management Science, Vol. 42 (4), pp. 541-558.

Burstein, M. L. (1961). "Measurement of Quality Change in Consumer Durables," The Manchester School Review of Economics and Social Studies, Vol. 29, pp. 267-279.

Gandal, N. (1994). "Hedonic price indexes for spreadsheets and an empirical test for network externalities," RAND Journal of Economics, Vol. 25, pp. 160-170.

Gandal, N. (1995). "Competing Compatibility Standards and network Externalities in the PC Software Market," Review of Economics and Statistics, Vol. 77, pp. 599-608.

Gordon, R. J. (1990) "The Measurement of Durable Goods Prices," The University of Chicago Press,

Griliches, Z. (1971). "Hedonic Price Indexes Revisited" in Zvi Griliches (ed.): Price Indexes and Quality Change: Studies in New Methods of Measurement, pp. 3-15. Cambridge, MA: Harvard University Press.

Mairesse, J. and B. Hall (1994). "Comparing the Productivity of Research and Development in French and United States Manufacturing Firms," mimeo.

Moch, D. (1995). "Ein Preisindex für PC-Datenbanksoftware - Eine empirische Untersuchung," in: D. Harhoff and M. Müller (eds.), Preismessung und technischer Fortschritt,. pp. 81-107. Baden-Baden: Nomos.

Moulton, B. R. (1996). "Bias in the Consumer Price Index: What Is the Evidence?," Journal of Economic Perspectives, Vol 10, No. 4, pp. 159-178.

Nelson, R. A., T. L. Tanguay, and C. D. Patterson (1994). "A Quality-Adjusted Price Index for Personal Computers," American Statistical Association, Vol. 12, pp. 23-31.

Muellbauer, J. (1974). "Household Production Theory, Quality and the 'Hedonic Technique'," American Economic Review, Vol. 64, pp. 977-994.

Rosen, S. (1974). "Hedonic Prices and Implicit Markets: Product Differentiation in Pure Competition." Journal of Political Economy, Vol. 82, pp. 34-55.

Szenzenstein, J. (1995). "Preisindizes für industrielle Güter in der amtlichen Statistik," in: D. Harhoff and M. Müller (eds.), Preismessung und technischer Fortschritt, pp. 11-36. Baden-Baden: Nomos.

Trajtenberg, M. (1990). Economic Analysis of Product Innovation: The Case of CT Scanners. Cambridge, Mass., Harvard University Press.

Triplett, J. E. (1971). "Quality Bias in Price Indexes and new Methods of Quality Measurement," in: Z. Griliches (ed.), Price Indexes and Quality Change, pp. 180-214. Cambridge, Mass.

Triplett, J. E. (1990). „Hedonic Methods in Statistical Agency Environments: An Intellectual Biopsy.“ in Berndt, E. R. and J. E. Triplett (eds.), Fifty Years of Economic Measurement, pp. 207-233. Chicago, University of Chicago Press.

White, H. (1980). "A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity." Econometrica, Vol. 48, 817-830.

Data Appendix

Average Prices, Standard Deviations and Frequencies by Year and Software Product

	Year									
Product	86	87	88	89	90	91	92	93	94	Total
Clipper	2130 (240) n=10	2291 (338) n=13	1712 (164) n=22	1607 (143) n=31	1571 (102) n=42	1541 (88) n=39	1493 (67) n=21	1536 (75) n=21	1522 (53) n=12	1642 (254) n=211
David	. . n=0	. . n=0	. . n=0	475 (32) n=3	652 (51) n=5	352 (167) n=4	. . n=0	. . n=0	. . n=0	508 (165) n=12
dBASE III	1449 (120) n=13	1463 (116) n=16	1386 (94) n=24	1370 (96) n=10	1447 (43) n=7	1480 (159) n=2	935 (50) n=5	. . n=0	. . n=0	1389 (159) n=77
dBASE IV	. . n=0	. . n=0	. . n=0	1569 (144) n=40	1520 (122) n=38	1550 (106) n=38	1362 (140) n=24	1358 (154) n=24	1090 (218) n=11	1467 (183) n=175
dBASE V	. . n=0	. . n=0	. . n=0	. . n=0	. . n=0	. . n=0	. . n=0	. . n=0	990 (59) n=5	990 (59) n=5
Foxbase	. . n=0	1106 (226) n=4	906 (273) n=10	1390 (295) n=8	1395 (369) n=9	932 (153) n=12	. . n=0	. . n=0	. . n=0	1124 (342) n=43
Foxpro	. . n=0	. . n=0	. . n=0	. . n=0	1673 (136) n=9	1434 (78) n=18	1396 (67) n=15	940 (172) n=9	860 (357) n=6	1323 (305) n=57
Paradox	. . n=0	1448 (235) n=4	1574 (64) n=14	1574 (91) n=21	1348 (384) n=31	733 (137) n=30	785 (166) n=16	798 (196) n=12	888 85) n=3	1141 (422) n=131
Total	1745 (388) n=23	1714 (500) n=37	1458 (302) n=70	1521 (232) n=113	1465 (285) n=141	1275 (389) n=143	1262 (292) n=81	1256 (320) n=66	1163 (321) n=37	1399 (358) n=711

Note: See text for an explanation of the data source and sampling procedure.