December 2005

# Software Versioning and Quality Degradation: An Exploratory Study of the Evidence

Anindya Ghose
*New York University*

Arun Sundararajan
*New York University*

# SOFTWARE VERSIONING AND QUALITY DEGRADATION: AN EXPLORATORY STUDY OF THE EVIDENCE

**Anindya Ghose and Arun Sundararajan**
Leonard N. Stern School of Business
New York University
New York, NY Y.S.A.
**aghose@stern.nyu.edu** **asundara@stern.nyu.edu**

## Abstract

*We present a framework for measuring software quality using pricing and demand data, and empirical estimates that quantify the extent of quality degradation associated with software versioning. Using a 7-month, 108-product panel of software sales from Amazon.com, we document the extent to which quality varies across different software versions, estimating quality degradation that ranges from as little as 8 percent to as much as 56 percent below that of the corresponding flagship version. Consistent with prescriptions from the theory of vertical differentiation, we also find that an increase in the total number of versions is associated with an increase in the difference in quality between the highest and lowest quality versions, and a decrease in the quality difference between neighboring versions. We compare our estimates with those derived from two sets of subjective measures of quality, based on CNET editorial ratings and Amazon.com user reviews, and we discuss competing interpretations of the significant differences that emerge from this comparison. As the first empirical study of software versioning that is based on both subjective and econometrically estimated measures of quality, this paper provides a framework for testing a wide variety of results in Information Systems that are based on related models of vertical differentiation, and its findings have important implications for studies that treat Web-based user ratings as cardinal data.*

**Keywords**: Software quality, vertical differentiation, price discrimination, quality distortion, information goods, Internet, electronic commerce, sales rank

## Introduction

This paper develops and estimates a model for assessing the relative quality of software versions using publicly available e-commerce pricing and demand data. It presents the first econometric assessment of the extent of quality degradation across a 108-product panel of software versions, using 7 months of pricing data from Amazon.com, and a new technique for inferring demand levels from reported sales ranks. These econometric estimates are contrasted with corresponding subjective assessments of quality degradation based independently on editorial ratings (from CNET) and consumer ratings (from Amazon.com), and competing explanations for the significant differences that emerge from this comparison are discussed.

Many manufacturers create product lines by first developing a flagship product with an optimal level of features and functionality, and then creating one or more inferior versions by deliberately reducing the quality of this flagship product. This practice is commonly referred to as *quality degradation*, and has been documented across a variety of industries (Deneckere and McAfee 1996). Such quality degradation is ubiquitous in the software industry. There are multiple versions of a large number of popular desktop software packages that differ only in their quality or number of features (rather than in their development or release date), and which are sold at different prices. At any point in time, one can find different versions of popular software titles like Adobe Acrobat, TurboTax, Microsoft Money, and Norton AntiVirus available. These are examples of software titles for which a firm has developed a flagship version, disabled a subset of the features or modules of this version, and released both the higher quality version and one or more lower quality versions simultaneously. Correspondingly, a large variety of software manufacturers make a limited functionality version of their product available for free, and charge a positive price for a full-featured version (Eudora and Eudora Light being a popular example). This prevalence is not surprising, given that the cost of disabling features or removing software modules is relatively low, as are the variable costs of producing software.

The theory used to study software versioning and quality degradation typically draws from second-degree price discrimination (Mussa and Rosen 1978). The key idea drawn from this theory is that of segmenting the market using multiple versions that are quality-differentiated substitutes, and that are created by strategically distorting down the quality of a flagship version. Many recent papers in information systems have used this theory to study the optimal number of versions for a seller of information goods (Bhargava and Choudhary 2001; Raghunathan 2000; Varian 2000), often concluding that a single version is optimal. Moreover, this underlying model of market segmentation using quality distortion is used in an increasing number of related IS studies that investigate, among other things, personalized pricing (Choudhary et al. 2005), optimal software upgrade paths (Bala and Carr 2004; Sankaranarayanan 2005), pricing of online services (Bhargava and Sundaresan 2003), efficiency and pricing of interorganizational (IOS) systems (Barua et al. 1991; Nault 1994) and managing digital piracy (Chellappa and Shivendu 2005; Oestreicher-Singer and Sundararajan 2004; Sundararajan 2004a).

To summarize, some notion of quality assessment by consumers, and of strategic degradation of quality levels by sellers, is embedded in a variety of IS studies that use the underlying model of vertical differentiation. Since each of these studies makes different managerial and policy prescriptions based on their models, it seems important to determine how to actually measure levels of quality and quality degradation predicted by such models toward setting up a way of testing their theories empirically and toward exploring whether quality degradation estimates based on the model of vertical differentiation seem reasonable. Moreover, since there are a growing number of Internet-based resources that report more subjective measures of software quality (editorial reviews and user reviews are the two most common), it is likely that there is some information about actual customer perceptions of quality differences between software versions contained in such ratings.

The central objective of our study is, therefore, to empirically estimate the measures of software quality and quality degradation predicted by the commonly used economic theory of vertical differentiation, to assess how they vary across software titles, and to contrast these estimates with those based on subjective Internet-based ratings. We do so by making the following contributions:

(1) We develop a method for directly estimating the extent of quality degradation based on the framework of price discrimination using vertical differentiation, and using publicly available pricing and demand data.

(2) We provide the first systematic estimate of the extent of quality degradation in the software industry, using a 7-month, 108-product panel of demand and pricing data gathered from Amazon.com.

(3) We contrast these economic estimates of quality degradation with two independent subjective assessments of software quality: editorial ratings gathered from CNET and average consumer ratings gathered from Amazon.com.

Our estimates of quality degradation across software versions indicate that, relative to the assessed quality of the flagship version, the quality levels of inferior versions are degraded from little as 8 percent to as much as 56 percent, and that the extent of degradation varies quite widely across software titles, and within sets of titles with two versions and three versions. Moreover, we find that an increase in the total number of versions is associated with an increase in the difference in quality between the highest and lowest quality versions, and a decrease in the quality difference between successive (or neighboring) versions. This is consistent with the predictions of the theory of vertical differentiation.

However, we also find that the economic estimates of quality degradation are significantly *different* (and significantly higher) than those assessed from subjective ratings. Put differently, when data about the actual purchasing behavior of customers is embedded into the economic model, it predicts very different levels of perceived quality differences than those suggested by the subjective ratings that these customers and other experts assign to the different versions of a software title. There are at least two possible interpretations of the differences we observe.

(1) The economic theory systematically models wider variations in software quality than are actually observed in practice. This may have important implications for the managerial and policy prescriptions derived from models that are based on this theory.

(2) The numbers or ratings that subjectively measure quality differences between software versions tend to systematically understate the actual differences, where by actual differences, we mean those based on economic measures of how much

quality affects consumer willingness to pay. In other words, these ratings, while having a reasonable ordinal interpretation, are not robust cardinal measures of quality.

Our study makes other research contributions as well. By transforming the parameters of a commonly used analytical model into those that can be estimated from demand data, in particular, to assess quality distortion and quality ratios across versions directly from demand and price data, we provide a new framework for future empirical studies of software versioning. In our concluding section, we discuss many directions for future research that might use this framework. We also report on a fairly comprehensive new method for converting Amazon sales rank data into demand data, which uses a combination of purchasing experiments and analysis of the ranking time series and provides the first such calibration for the computer software industry. Moreover, our analytical model is developed in a manner that enables one to estimate customer distribution characteristics from widely available demand data in a straightforward way. This makes future empirical studies of pricing and quality differentiation in other IT industries more easily feasible. Thus, our paper also adds to the new emerging stream of literature that has used e-commerce-based panel data to conduct industry specific studies (Ghose et al. 2005) and new instances of existing phenomena such as auctioneer–bidder strategies and price formation in online auctions (Bapna et al. 2004).

The preceding discussion has highlighted a fraction of the IS literature that is related to our current paper in its approach to modeling quality degradation. To our knowledge, this is the first paper that attempts to empirically validate this modeling approach using data in the software industry. Research that has assessed quality degradation in other industries include studies of the airline industry (Borenstein and Rose 1994) and the cable television industry (Crawford and Shum 2005), although neither of these papers contrast econometrically estimated quality levels with subjective measures. Additionally, there is an impressive body of literature on software quality (for instance, Slaughter et al. 2000). Much of this literature is empirical, although our paper differs from others on two important dimensions. First, other research tends to study quality issues for large-scale specialized software implementations in organizations, rather than measuring quality for mass market shrink-wrapped software. Second, that research tends to assess and study software quality using supply side measures—intrinsic measures such as reliability and integrity of the source code, and the number of defects per function point—while our approach is focused on different demand-side measures of software quality.

The rest of this paper is organized as follows. The next section presents our analytical model, which relates pricing and quality degradation to customer characteristics, and describes how to connect the equations derived from this model to our demand data. We then describe our data set, our method for converting sales rank data into demand data, and some details on how we estimate our model's key parameters. The results of our estimation of quality degradation in the software industry are presented and contrasted with the subjective measures of software quality. The final section presents conclusions and outlines directions for future research.

## Model

A monopolist sells $n$ versions of a software product. This seller first develops the highest quality (or flagship) version of quality $s_1$, and then degrades the quality of this version to create a set of inferior substitutes, of quality $s_2, \ldots, s_n$, where $s_1 > s_2 > \ldots > s_n$. The price charged by the seller for version $i$ is denoted $p_i$.

Customers are modeled as varying in their preferences for quality. A customer of type $\theta \in \Theta$ is willing to pay up to $U(s, \theta)$ for a version of quality $s$, where $U(s, \theta)$ is non-decreasing in both its arguments. The set $\Theta$ is discrete, with elements $\theta_i$. Customer types are distributed according to a probability measure $F$ over $\Theta$, and for notational convenience, we denote the measure of customers of type $\theta_i$ as $f_i \equiv F(\theta_i)$. We make the assumption of discrete types for subsequent ease of estimation (more on this later). Our analysis could equivalently assume that customer types are uniformly distributed over some continuous interval.

Since the versions are substitutes, each customer purchases up to one version. A customer of type $\theta$, therefore, purchases a version of quality $s_i$ if version $i$ yields the highest positive level of consumer surplus, that is if

$$U(s_i, \theta) - p_i > U(s_j, \theta) - p_j \tag{1}$$

for each $j \neq i$, and if

$$U(s_i, \theta) - p_i \geq 0. \tag{2}$$

The seller's problem is to choose the optimal number of versions, the quality level for each version, and their associated prices. We assume that the utility function takes the following simple quadratic form:[1]

$$U(s,\theta)=\begin{cases}\theta s-\frac{1}{2}s^2,\ s<\theta\\ \frac{1}{2}\theta^2,\ s\geq\theta\end{cases}. \tag{3}$$

Equation (3) indicates that customers value increasing quality at a diminishing rate, and the highest quality level that a customer of type $\theta$ is interested in is $s = \theta$. Therefore, if $\Theta = \{\theta_1, \theta_2, \ldots, \theta_n\}$, the socially optimal outcome involves the seller offering $n$ versions, with quality levels $s_1 = \theta_1$, $s_2 = \theta_2$, and so on. We analyze the cases of two types, that is, $\Theta = \{\theta_1, \theta_2\}$ and three types, that is, $\Theta = \{\theta_1, \theta_2, \theta_3\}$, because our data set contains software titles with either two or three versions.

### *Two Versions*

We start by assuming that $\Theta = \{\theta_1, \theta_2\}$. The seller offers the flagship version of quality $s_1$, and may offer a second version of quality $s_2 < s_1$. The standard method of analysis (see, for example, Sundararajan 2004a) uses the revelation principle to ensure that the seller only needs to consider direct mechanisms and will design one quality-price pair for each type, such that these pairs are incentive-compatible (IC) and individually rational (IR). These conditions yield the following price equations:

$$p_2 = U(s_2,\theta_2), \tag{4}$$

$$p_1 = U(s_1,\theta_1) - U(s_2,\theta_1) + p_2. \tag{5}$$

The firm, therefore, chooses $s_1$ and $s_2$ to maximize

$$f_1[U(s_1,\theta_1) - U(s_2,\theta_1)] + U(s_2,\theta_2), \tag{6}$$

and maximizing (6) yields the optimal quality levels

$$s_1 = \theta_1, \tag{7}$$

$$s_2 = \max\left\{\theta_2 - \frac{f_1}{f_2}(\theta_1-\theta_2), 0\right\}. \tag{8}$$

Equations (7) and (8) indicate that the seller will offer two versions so long as the difference between the quality preferences of the two types is not too large, and there is a sufficient fraction $f_2$ of lower type customers. Notice that the flagship version is assigned the socially optimal quality level, while the quality of the lower version is distorted downward. In this case, the corresponding prices as functions of the model's primitives are

$$p_1 = \frac{f_1}{f_2}(\theta_1-\theta_2)^2 + \frac{1}{2}\left[\theta_2^2 + (\theta_1-\theta_2)^2\right], \tag{9}$$

$$p_2 = \frac{1}{2}\left[\theta_2^2 - \frac{f_1^2}{f_2^2}(\theta_1-\theta_2)^2\right]. \tag{10}$$

We refer back to these expressions shortly to discuss how we use them to estimate quality.

---

[1]A more commonly used functional form is even simpler: $U(s, \theta) = \theta s$, although that predicts that for a good with constant variable costs, a single version is optimal, which, based on our data, is clearly inconsistent with the software industry.

### *Three Versions*

Next, we assume that $\Theta = \{\theta_1, \theta_2, \theta_3\}$. The seller will, therefore, offer the flagship version of quality $s_1$, along with up to two other versions with quality $s_2 < s_1$ and $s_3$ $s_2$. The corresponding price equations from the IC and IR conditions are

$$p_3 = U(s_3, \theta_3), \tag{11}$$

$$p_2 = U(s_2, \theta_2) - U(s_3, \theta_2) + p_3, \tag{12}$$

$$p_1 = U(s_1, \theta_1) - U(s_2, \theta_1) + p_2. \tag{13}$$

The firm chooses $s_1$, $s_2$, and $s_3$, to maximize

$$f_1[U(s_1, \theta_1) - U(s_2, \theta_1)] + (f_1 + f_2)[U(s_2, \theta_2) - U(s_3, \theta_2)] + U(s_3, \theta_3), \tag{14}$$

yielding the following optimal quality levels:

$$s_1 - \theta_1 \tag{15}$$

$$s_2 = \max\left\{\theta_2 - \frac{f_1}{f_2}(\theta_1 - \theta_2), 0\right\}. \tag{16}$$

$$s_3 = \max\left\{\theta_3 - \frac{f_1 + f_2}{f_3}(\theta_2 - \theta_3), 0\right\}. \tag{17}$$

Assuming that all of these quality levels are in fact non-zero, the corresponding expressions for prices as a function of the model's basic parameters are

$$p_1 = \frac{f_1}{f_2}(\theta_1 - \theta_2)^2 + \frac{f_1 + f_2}{f_3}(\theta_2 - \theta_3)^2 + \frac{1}{2}[\theta_3^2 + (\theta_2 - \theta_3)^2 + (\theta_1 - \theta_2)^2], \tag{18}$$

$$p_2 = \frac{f_1 + f_2}{f_3}(\theta_2 - \theta_3)^2 + \frac{1}{2}[\theta_3^2 + (\theta_2 - \theta_3)^2 - \frac{f_1^2}{f_2^2}(\theta_1 - \theta_2)^2], \tag{19}$$

$$p_3 = \frac{1}{2}\left[\theta_3^2 - \left(\frac{f_1 + f_2}{f_3}\right)^2 \frac{(\theta_1 - \theta_2)^2}{2}\right]. \tag{20}$$

### *Linking this Theory to Prices and Demand Data*

Rather than the numerical values of quality implied by the model, we are interested in the extent of quality degradation for different products, that is, in the ratios $s_1/s_j$ for each $j > 1$. Given a set of price data for each of the versions of a software product, one needs estimates of each of the $\theta_i$ and $f_i$ parameters in order to use equations (7) and (8) or (15), (16), and (17) to compute these quality ratios. Further, in the underlying model, notice that the demand for version *i* is simply $f_i$.

These observations lead to a natural way of linking the theoretical model to a data set of prices and demand. First, for each product, we observe the fraction of demand realized by each of its versions, in each of our time periods, and use this as an observation about the corresponding $f_i$. We use these observations to assess a maximum likelihood estimate of each $f_i$ for each product (further details follow a description of our data). Given these estimates, we can use observed average prices and the system of equations (18), (19), and (20) to estimate $\theta_1$, $\theta_2$, and $\theta_3$ (or correspondingly, equations (9) and (10) in the two-version model to estimate $\theta_1$ and $\theta_2$). Notice that, given prices and the $f_i/f_j$ ratios, (18), (19) and (20) is a system of three equations in three unknowns. These estimates can then be used in equations (15), (16), and (17) or in equations (7) and (8) to estimate the quality degradation associated with versions of the software title.

## Data and Estimation

We estimate our models using a panel data set compiled from publicly available information about software prices and sales rankings, gathered using automated Java scripts to access and parse HTML and XML pages downloaded from Amazon.com. The panel includes over 280 products, with an equal number from each of five major categories, Business and Productivity, Graphics and Development, Security and Utilities, Children's Software, and Operating Systems. These are major categories listed by Amazon.com, and resemble a parallel categorization by CNET.com, although we do not use this categorization in any substantive way.[2]

Of our 280 products, we identify 108 as belonging to a family of different versions of the same product. In this context, it is important to distinguish between *versions* and successive *generations*. For instance, Adobe Standard 7.0 and Adobe Professional 7.0 are two different versions of Adobe Acrobat 7.0. Similarly, TurboTax Premier 2004, Deluxe 2004, and Standard 2004 are three different versions of TurboTax 2004. On the other hand, Adobe Illustrator 10.0 and Adobe Illustrator CS are successive *generations* of Illustrator and, while substitutes, were developed at different points in time and released over two years apart. As a consequence, this pair of products is not consistent with our underlying economic model of versioning, in which a seller develops a flagship version and then strategically degrades its quality to create inferior versions.

We separate our products into two sets. The first consists of all software titles which have three different versions. The ordering of these versions is naturally inferred from their titles (a typical labeling would be Premier, Deluxe, and Standard, for instance, in decreasing order of quality). The second category consists of all products that have two versions (often labeled Professional and Standard, in decreasing order of quality). We end up with 27 software titles with two versions, and 18 software titles with three versions (for a total of 108 versions).

We collected data every 8 hours, over a 7-month period (from November 2004 to May 2005). Each observation contains the product's list price, its Amazon retail price, its Amazon sales rank (which serves as a proxy for units of demand, as described further later), the date the product was released into the market, the average customer rating for the product, the number of reviews based on which the average rating was computed, and some secondary market data. The summary statistics of our data are in Table 1.

For benchmarking purposes, we have also collected similar data from Buy.com: sales ranks, list prices, retail prices, and so on. Similar to Amazon.com, Buy.com provides sale rankings of all of its products publicly and these sales ranks are also based on actual quantities sold at their site. The Buy.com data exhibits qualitatively similar characteristics as the Amazon.com data, and since we do not use this data further in our analysis, it is not described.

**Table 1. Summary Statistics of Our Data**

| Variable | Mean | Standard Deviation | Minimum | Maximum |
|---|---|---|---|---|
| Sales Rank | 1649.61 | 1971.26 | 1 | 11622 |
| List Price | 99.2 | 226.2 | 19.99 | 1799.99 |
| Amazon Price | 95.53 | 208.57 | 10.95 | 1699.99 |
| Customer Rating | 3.14 | 0.99 | 1 | 5 |
| Number of Reviewers | 56.2 | 81.6 | 7 | 606 |

[2]We do not include the entertainment segment of the software market since it is characterized by the concurrent availability of successive generations of a given product instead of quality degraded versions.

### *Inferring Software Demand from Sales Ranks*

A few recent papers have used the following Pareto relationship to infer product quantities from Amazon.com sales ranks:

$$q = \delta(SalesRank)^{\beta} \tag{21}$$

Chevalier and Goolsbee (2003) estimate the parameters of this equation for books by associating demand data with sales rank on *The Wall Street Journal* best-seller list, and by independently conducting a purchasing experiment on one book, whose actual weekly demand was known to them, and observing the extent to which its sales rank reacted to their purchases. They estimate the value of $(1/\beta)$ to be –1.2. Brynjolfsson et al. (2003) provide an alternative estimate of the parameters of equation (21) for books, using data from a book publisher that maps observed sales rank to the number of copies the publisher sold to Amazon, and estimate $\beta = -0.871$ (this is the parameter $\beta_2$ in their model), $\log[\delta] = 10.526$ (this is the parameter $\beta_1$ in their model).

To our knowledge, there are no corresponding estimates available for software, and industry-specific demand patterns preclude using estimates from book demand for the software industry. Moreover, in summer 2004, Amazon altered its sales rank system in the following way: they eliminated their three-tier system, updating ranks each hour for most products (rather than merely for the top products), and they moved to a system that uses exponential decays to give more weight in the sales rank to newer purchases.

We, therefore, conducted an independent analysis to convert our measured sales ranks into demand data. We retain the assumption of a Pareto relationship (21) between demand and sales rank. We combined an analysis of a 2-week sales rank time series for each of our products with a set of purchasing experiments to relate movements in sales rank to unit demand, and used these results to estimate the following OLS equation:

$$\log[q+1] = \log[\delta] + \beta \log[rank], \tag{22}$$

where $q$ is average weekly demand and *rank* is the corresponding average salesrank[3] The results of this estimation are summarized in Table 2. To provide a sense for what this estimate implies, weekly sales of two units correspond to an average salesrank of about 3,100, weekly sales of 10 units correspond to an average salesrank of about 440, and weekly sales of 25 units correspond to an average salesrank of about 150.

### *Estimating the Customer Type Distribution*

The preceding experiment enables us to associate our sales ranks with corresponding periodic unit demand levels. Now, consider a software title with $n$ versions, and demand data over T periods. In any period $t$, let the demand for version $i$ be $q_{it}$, and define the total demand for this title during period $t$ as $q_t = \sum q_{it}$. One can model this demand as the result of $q_t$ draws from the distribution $F$ over $\Theta$, with the outcome reflecting $q_{it}$ draws of type $\theta_i$, for each version $i$ of each software title.

**Table 2. Mapping Average Sales Rank to Unit Demand for Software**

| Variable | Estimated Value |
|---|---|
| log[d] | $8.352^{***}(0.042)$ |
| β | -0.828***(0.032) |
| $R^2$ | 0.779 |

***significant with $p \leq 0.001$

[3]Similar to Brynjolfsson et al. (2003), we used White's heteroskedasticity-consistent estimator (see Greene 2000, p. 463) to estimate both parameters. Details of the experiment are available in Center for Digital Economy Working Paper CeDER-05-20 (Ghose and Sundararajan 2005) at **http://www.stern.nyu.edu/ceder/**.

Define $Q_i$ as the random variable that takes the value 1 if a draw from $F$ yields $\theta_i$, and takes the value *0* otherwise. It follows that $Q_i$ is a Bernoulli random variable with parameter $f_i$. Therefore, each realization of a unit of demand for any version of the product is an observation about the true value of $f_i$, and it is well known that with $n$ such observations, the maximum likelihood estimator of $f_i$ is simply the fraction of true realizations. As a consequence, once we have computed the periodic demand levels for each version of each title, the maximum likelihood estimate of $f_i$ for a specific software title is simply

$$f_i = \left(\sum_{t=1}^{T} q_{it}\right) \Bigg/ \left(\sum_{t=1}^{T}\sum_{k=1}^{n} q_{kt}\right) \tag{23}$$

or the estimated demand for the version as a fraction of the total demand for all versions of the title. As described earlier, once we have estimates of $f_i$ for each version $i$ of each product, we are able to compute the implied corresponding values of $\theta_i$ and the corresponding quality degradation levels.

It is worth noting that our equations always involve a ratio of two $f_i$ values (rather than an $f_i$ value in isolation). Therefore, if one chooses the appropriate periodic demand rate associated with an average sales rank, these ratios can be computed directly from average sales ranks, since, based on equations (21) and (23) and an appropriate normalization for the length of the time interval which cancels out in the numerator and denominator, this simplifies to

$$\frac{f_i}{f_j} = \left(\frac{rank_i}{rank_j}\right)^{\beta}, \tag{24}$$

where $rank_i$ is the average sales rank of product $i$.

## Evidence

Before we present the results of our estimated quality degradation, and contrast them with the subjective measures we have collected, it seems important to distinguish between quality degradation and quality distortion, since the latter term is used quite extensively in the price discrimination literature. Our measure of *quality degradation* for any version is simply the ratio of the estimated quality of the highest version to the estimated quality of the version in question. For instance, the extent of quality degradation for the second-highest quality version of a product with three versions is $s_1/s_2$, which based on equations (15) and (16), is

$$\frac{s_1}{s_2} = \theta_1\left(\theta_2 - \frac{f_1}{f_2}(\theta_1 - \theta_2)\right)^{-1} \tag{25}$$

A higher value of this ratio implies a higher difference in quality and, therefore, more significant quality degradation. On the other hand, *quality distortion* refers to the extent to which the quality of an inferior version $i$ has been distorted below the socially optimal level $\theta_i$. We also report on our estimated percentage quality distortion levels, which are simply $(\theta_i – s_i)/\theta_i$, although we do not discuss them much. For instance, the percentage of quality distortion for the second-highest quality version of a product with three versions, based on equations (15) and (16), is

$$1 - \frac{s_2}{\theta_2} = \frac{f_1}{f_2}\left(\frac{\theta_1}{\theta_2} - 1\right) \tag{26}$$

### *Estimated Quality Degradation*

For titles with two versions, we find that the quality ratios vary from as low as 1.09 to as high as 1.75. This reflects a downward degradation in the quality of the flagship version from as little as 8 percent to as much as 43 percent, with a mean degradation of about 27 percent. For titles with three versions, we find that the quality ratios for the medium quality version (that is, the ratios $s_1/s_2$) range from 1.08 to 1.46, thereby reflecting degradation of the quality of the flagship version from as low as 7 percent to as high as 31 percent (with a mean of about 21 percent). The corresponding quality ratios for the low quality version (the ratios $s_1/s_3$)

are between 1.63 to 2.31, which correspond to quality degradation ranging from 39 percent to as much as 57 percent (and mean of about 44 percent).[4]

These estimates indicate that when a product line has three versions, the extent of quality degradation between the best version and the second-best version is significantly lower (both on average and in its variance) than the extent to which the quality of a second-best version is degraded when a third version does not exist. However, the extent to which the quality of the lowest version is degraded when the product line has three versions is significantly more than when the product has just two versions. We verify these statements by testing the difference in mean between both pairs of data (finding significant t-statistics in each case). These results are interesting because they are consistent with what the theory of vertical differentiation would predict. All else being equal, an increase in the number of versions offered will increase the extent of quality degradation of the lowest quality version, but will also reduce the differences in quality between neighboring versions.

Exactly the same statements can be made about the percentage of quality distortion (highest for lowest of three versions, lowest for second of three versions, significant differences in means). We do not explore any welfare issues using these estimates, although this represents an interesting direction for future work.

### *Contrasting Economic and Subjective Measures of Quality Degradation*

We next report on our estimates of quality degradation based on two subjective measures of assessed quality, from CNET and from Amazon.com. A set of editors at CNET evaluate most software products according to a standard set of review criteria, and rank these products on a scale of 1 to 10. According to CNET, they judge a product on the quality and appropriateness of its features set along with service and support provided by the firm. They also evaluate the number and severity of any bugs as well as the overall ease of setup, configuration, and use. We use these summary scores from CNET as our first subjective measure of quality, and assess quality degradation by computing the ratio of scores for different versions of a title. Prior studies have used such rankings as an objective measure of software quality (for instance, Liebowitz and Margolis 1999). We collected these scores from CNET's Web site on a periodic basis. Since CNET also archives ratings for older products, we have been able to gather these ratings for most products in our dataset.

Our second source of subjective quality assessments is from reviews for each product provided by Amazon.com's customers. Each review contains a written report as well as a numerical score on a scale of 1 to 5. We use the average numerical score associated with a product as our second subjective measure of quality. We collected longitudinal data on these ratings, along with the total number of reviewers on which the average rating is based. We dropped product ratings which were based on reviews by five or fewer customers. The average number of reviews for the remaining products is 56, and the number of reviews ranges from 7 to 605 (most have 20 to 50 reviews).

We find that the extent of quality degradation assessed from our economic estimates is significantly higher than those assessed from the subjective measures of quality. The mean quality degradation is significantly *higher* for comparisons of $s_1/s_2$ for products with two versions and with three versions, and for comparisons of $s_1/s_3$ for the subjective measures based both on CNET editorial ratings and on Amazon customer reviews. These differences are somewhat higher for CNET than for Amazon. Furthermore, the differences were most stark when comparing the extent of quality degradation of the lowest quality version for products with three versions.

There are many ways in which one might interpret these findings. One interpretation might be that in models of vertical differentiation, the extent to which quality varies across versions in the model are far wider than are actually observed in practice. In other words, the extent of the optimal quality difference prescribed by the model's quality parameters $s_i$ and $s_j$ may be higher than the actual quality difference that is required to obtain the appropriate optimal magnitude in value difference; the latter difference is what influences the willingness to pay of customers and the firm's eventual success with price discrimination based on versioning. This would suggest that prescriptions from models of versioning or price discrimination that are based on the magnitude of the quality difference across versions should be interpreted carefully.

[4]Estimates are available in Ghose and Sundararajan 2005.

**Table 3. Summary of Difference in Mean Tests between Quality Degradation Levels**[†]

| | $s_1/s_2$ for models with two versions | | $s_1/s_2$ for models with three versions | | $s_1/s_3$ for models with two versions | |
|---|---|---|---|---|---|---|
| | Model | CNET | Model | CNET | Model | CNET |
| *Mean* | 1.38 | 1.07 | 1.25 | 1.03 | 1.79 | 1.10 |
| *Variance* | 0.03 | 0.01 | 0.01 | 0.002 | 0.03 | 0.01 |
| *Observations* | 23 | 23 | 16 | 16 | 17 | 17 |
| *t-statistic* | **7.93** | | **8.59** | | **14.54** | |
| *p* | 5E-09 | | 9E-09 | | 1E-12 | |
| | **$s_1/s_2$ for models with two versions** | | **$s_1/s_2$ for models with three versions** | | **$s_1/s_3$ for models with two versions** | |
| | Model | Amazon | Model | Amazon | Model | Amazon |
| *Mean* | 1.37 | 1.03 | 1.25 | 1.14 | 1.75 | 1.29 |
| *Variance* | 0.02 | 0.03 | 0.01 | 0.02 | 0.02 | 0.02 |
| *Observations* | 19 | 19 | 11 | 11 | 12 | 12 |
| *t-statistic* | **6.25** | | **2.01** | | **8.13** | |
| *p* | 2E-07 | | 0.03 | | 3E-08 | |

[†] We estimated from our data, from CNET's editorial ratings, and from Amazon's average customer ratings, respectively. The numbers under the label "Model" refer to our estimates.

Another interpretation might be that the numbers that subjectively measure quality differences between software versions tend to systematically understate the actual differences, where by actual differences we mean those based on economic measures of how much quality affects consumer willingness to pay. These subjective ratings might, therefore, be a good way of ranking different versions, but their numerical magnitudes may not be appropriate cardinal measures of quality. This interpretation has important implications for future research, because editorial ratings have been used as measures of software quality in prior studies, and aggregate customer feedback measures from eBay, Amazon.com, and various other review sites are frequently used in IS research as cardinal measures of some form of quality in studies of seller reputation, movie quality, used-good quality, and so on.

A third interpretation might simply be that editors and customers have a different benchmark when assessing the quality of different versions, and that these benchmarks (or reference points) are affected by what the customer or editor expects from a specific version. For example, a rating of 5 on a Professional version might require a higher level of overall quality than a rating of 5 for a Standard version. This would cause a systematic overstatement of the quality of lower versions as measured by these average ratings or reviews, which in turn would lead to lower assessed quality degradation levels.

A preliminary analysis toward a better understanding of the relationship between these objective and subjective measures did not yield results that were significant enough to report. Determining which of these interpretations might be the most valid remains an open question, though we believe that more data is required to answer this well.

## Conclusions and Directions for Future Research

This paper has presented the first empirical study of versioning in the software industry. The contributions of this study are summarized below.

(1) In order to assess the success of a chosen versioning strategy relative to others, it is useful for firms to derive an economic measure of the relative quality of each version that has been created based on quality degradation. This represents a considerable challenge in the software industry, because while subjective assessments of software value from independent experts and from its end users are available, there are no natural objective measures of product size or quality (counting the number of features is not really sensible, for instance). Therefore, objective assessments of software quality based on economic demand-side measures of a product's quality can be of managerial value. We develop a framework for directly

estimating the extent of software quality degradation based on a widely used model of price discrimination using vertical differentiation, and that can be estimated using pricing and demand data that is publicly available.

(2) We provide the first systematic estimate of the extent of quality degradation associated with versioning in the software industry. We do so by compiling and using a 7-month, 108-product panel of demand and pricing data for software sold on Amazon.com. Our results indicate that there is significant quality degradation associated with software versioning, and significant variations in its extent across software titles. Our estimates are consistent with theoretical predictions that an increase in the number of versions is associated with an increase in the quality difference between the highest and lowest quality version, but a reduction in the quality differences between neighboring versions.

(3) We provide new estimates of quality degradation between versions using two independent sources of subjective quality assessments: editorial ratings gathered from CNET and average user ratings gathered from Amazon.com. We contrast these estimates of quality degradation with those from our economic model. We show that the estimates of quality degradation from the latter are significantly and consistently higher than those assessed from subjective measures of software quality, and discuss different interpretations of this measured difference.

(4) We extend existing methods for imputing demand from Amazon.com's sales rank information, and provide the first calibration of this relationship for the software industry.

Apart from providing a first step toward testing other existing IS theories that are based on models of vertical differentiation, our work suggests a number of new directions for future research, and provides an infrastructure that can be used to explore these directions. A natural question that arises from our study is whether software versioning is in fact an optimal strategy for sellers, and if so, measuring the extent to which it increases profits. It is likely that the benefits from versioning are related to both the category of software and the extent to which the flagship version has been degraded to create each inferior version. Examining this relationship could be of particular interest to IS practitioners making pricing and product management choices.

We have also provided the first estimates of the extent of quality distortion for software (relative to the socially optimal quality level of a version). This is a first step toward assessing the magnitude of the welfare losses that ensue on account of this distortion. However, there are likely to be welfare gains from the prevalence of versioning, due to the expansion of the set of customers who can afford a version of the product. Comparing the relative gains and losses from quality distortion, given that the absence of this kind of distortion would lead to higher prices, represents another promising line of research. A related study might examine whether there is a relationship between measured quality distortion and subjective measures of quality, based on the hypothesis that subjective ratings assess product quality relative to a benchmark for that kind of version, rather than relative to the flagship version, and therefore might measure distortion rather than degradation. An analysis of the text associated with editorial reviews might be instructive in this regard.

A preliminary exploration of whether there are variations in the differences between subjective and economic measures of quality degradation across each of our product categories did not yield significant results, although this may be a consequence of the fact that there are insufficient titles in each category for any systematic differences to show up. For instance, it may be relatively straightforward for consumers and experts to assess the quality of finance and accounting software based on their features and ease of installation. However, the quality of security software is much harder to assess, since it is contingent on future performance at detecting and suppressing viruses, minimizing the probability of a breach, or detecting an intrusion. Studying this in more detail, using a larger data set, or perhaps longitudinal data, seems like another interesting direction for future work.

Finally, while our study has been of the software industry, many of our techniques can generalize to other IT industries. Future empirical researchers might use our method to map sales ranks to demand for other categories of products, which would facilitate new industry-specific quality degradation studies that answer related strategic and welfare questions in other IT product industries. We hope that our study is a first step in this direction.

## *References*

Bala, R., and Carr, S. "Pricing and Market Segmentation with Software Upgrades," Anderson School of Management, University of California, Los Angeles, 2004 (available online at **http://www.anderson.ucla.edu/documents/areas/fac/dotm/bio/pdf_SC09.pdf**).

Bapna, R., Goes, P., and Gupta, A. "Replicating Online Yankee Auctions to Analyze Auctioneers' and Bidders' Strategies," *Information Systems Research* (14:3), 2003, pp. 244-268.

Barua, A., Kriebel, C., and Mukhopadhyay, T. "An Economic Analysis of Strategic Information Technology Investments," *MIS Quarterly* (15:3), 1991, pp. 313-331.

Bhargava, H., and Choudhary, V. "Information Goods and Vertical Differentiation," *Journal of Management Information Systems* (18), 2001, pp. 89-106.

Bhargava, H., and Sundaresan, S. "Contingency Pricing for Information Goods and Services under Industry-Wide Performance Standards," *Journal of Management Information Systems* (20:2), 2003, pp. 113-136.

Borenstein, S., and Rose, N. "Competition and Price Dispersion in the US Airline Industry,". *Journal of Political Economy* (102), 1994, pp. 653-683.

Brynjolfsson, E., Hu, Y., and Smith, M. "Consumer Surplus in the Digital Economy: Estimating the Value of Increased Product Variety," *Management Science* (49), 2003, pp. 1580-1596.

Chevalier, J., and Goolsbee, A. "Measuring Prices and Price Competition Online: Amazon and Barnes and Noble," *Quantitative Marketing and Economics* (1), 2003, pp. 203-222.

Chellappa, R., and Shivendu, S. "Managing Piracy: Pricing and Sampling Strategies for Digital Experience Goods in Vertically Segmented Markets," *Information Systems Research*, 2005 (forthcoming).

Choudhary, V., Ghose, A., Mukhopadhyay, T., and Rajan, U. "Personalized Pricing and Quality Differentiation," *Management Science* (51:7), 2005, pp. 1120-1130

Crawford, G., and Shum, S. "Monopoly Quality Degradation and Regulation in Cable Television," Working Paper, University of Arizona, 2005.

Deneckere, R., and McAfee, P. "Damaged Goods," *Journal of Economics and Management Strategy* (5), 1996, pp.149-174.

Ghose, A., and Sundararajan, A. "Software Versioning and Quality Degradation? An Exploratory Study of the Evidence," Working Paper CeDER-05-20, Center for Digital Economy, Stern School of Business, New York University, 2005.

Ghose, A., Telang, R., and Krishnan, R. "Impact of Electronic Secondary Markets on Information Goods Supply Chain, *Journal of Management Information Systems* (22:2), 2005, pp. 91-120.

Greene, W. H. *Econometric Analysis*, Prentice-Hall, Upper Saddle River, NJ, 2000.

Liebowitz, S., and Margolis, J. *Winners, Losers and Microsoft: Competition and Antitrust in High Technology* (2nd ed.), Independent Institute, Oakland, CA, 1999.

Mussa, M., and Rosen, S. "Monopoly and Product Quality," *Journal of Economic Theory* (18), 1978, pp. 301–317.

Nault, B. "Quality Differentiation and Adoption Costs: The Case for Inter-Organizational Information System Pricing," *Annals of Operations Research* (71), 1997, pp. 115-142.

Oestreicher-Singer, G., and Sundararajan, A. "Are Digital Rights Valuable? Theory and Evidence from the eBook Industry," in *Proceedings of the 25th International Conference on Information Systems*, R. Agrawal, L. Kirsch, J. I. DeGross (Eds.), Washington, DC, December 2004, pp. 617-629.

Raghunathan, S. "Software Editions: An Application of Segmentation Theory to the Packaged Software Market," *Journal of Management Information Systems* (17:1), 2000, pp. 87-114.

Sankaranarayanan, R. "Excessive Software Upgrades: Reasons and Remedies," Social Science Research Network, June 9, 2005 (available online at **http://ssrn.com/abstract=764667**).

Slaughter, S., Harter, D., and Krishnan, M. S. "Effects of Process Maturity on Quality, Cost and Cycle time in Software Product Development," *Management Science* (46), 2000, pp. 451-466.

Sundararajan, A. "Managing Digital Piracy: Pricing and Protection," *Information Systems Research* (15), 2004a, pp. 287-304.

Sundararajan, A. "Nonlinear Pricing of Information Goods," *Management Science* (50), 2004b, pp. 1660-1673.

Varian, H. "Buying, Selling and Renting Information Goods," *Journal of Industrial Economics* (58), 2000, pp. 473-488.