

5% Solution To Pricing Software IP

BY PAUL BUTTEN*



An empirical, practical method developed to price shell-wrapped software by taking percent of installed replacement cost; calculated by data

For many organizations, software licensing represents an increasing share of their intellectual property portfolio. Because most software is new and the market uncertain, it would be useful to have a quick, "back-of-the-envelope" method of estimating the selling price. Existing methods of valuation generally do not work well for software, because software creation and development do not follow the standard "manufacturing" economic model, which often involves large capital expenditures for prototypal development, manufacturing labor and equipment, and raw materials. Software development does not require those additional capital costs. All an entrepreneur needs to get into business is a computer, a compact-disk (CD) pressing machine, and a web page.

The literature offers two little-or-product pricing^{1,2} and essentially nothing on software pricing or the application of standard valuation methods to software.³ The standard four valuation methods are inadequate for pricing software. The replacement method, a "backward-looking" approach to valuing products on the basis of total fixed expenditures, tends to grossly overprice the value of software, because most of the research is directed at solving some "big" problem, rather than specifically at creating software per se. The other three methods — industry royalty standards, the 25% profit rule, and net present value — are "forward-looking" techniques; in these approaches, it is assumed that a product is very close to market so that

profit margins, market penetration, sales revenues, etc. can be estimated. For new and unique software, these methods are often merely guesses or assumptions, "documented" only in the sense that they are written on paper.

PRICE IS IMPORTANT

Standard methods of product valuation fail to provide a "price" for software. Many licensing executives are skilled at recommending royalty percentages, say 15%, or providing platitudes such as "Get all you can get!" Such advice is worthless to an entrepreneur or organization trying to sell software directly.

Determining price is a critical issue for anyone selling software directly, or for the licensee who asks the licensee to set the price. If one is selling software, it is important to determine:

1. Is the licensee setting a reasonable price?
2. When holding an equity position, do the projected sales make sense on the basis of their product price?
3. As a cross-check on negotiated royalties, do the projected costs and profits add up to the list price?

THE MODIFIED REPLACEMENT COST

If no competing products existed, the most logical place to start in valuing a firm piece of software would be to determine the replacement cost (RC), in most cases, the total project cost or RC under which the software was created is known by the researcher. If the R&D work extended over a long time, the researcher's efforts probably were directed to pursue targets and dead ends, and the RC overestimates what it took to create the

software. One could ask the author what effort was necessary to create the software, but a better question to pose might be, "Knowing what you now know, how long would it take to recreate this software?" This question is more realistic because it is entirely application-driven, eliciting the cost to recreate the product directly (the "modified replacement cost," MRC).

The MRC is more representative than the RC. The author could be hired by a private company to rewrite the software, and a rewrite may not infringe on the existing copyright. Given the rapid pace of technological change, it is also reasonable to expect that the software could be rewritten today in a more efficient manner, using recently updated commercial software or hardware.

The MRC applies to "atomic-wrapped" software, which is defined as software that is essentially ready for commercialization. It comes with a manual that includes examples, offers relatively simple input and output operations, and runs on commonly available computers (such as personal computers or workstations). Software that lacks a manual, is written in exotic computer languages, requires advanced supercomputers for use, etc. is considered incomplete and not yet commercially viable; any pricing valuation should be delayed, or significantly discounted, until these concerns are addressed.

To date, no staff members have had to recreate their software, and their estimates are taken at face value. In fact, project managers, however, add a contingency factor. They know that programs tend to take longer than expected. Chas-

*Senior Account Executive, Software Licensing Manager, Argonne National Laboratory, Argonne, Illinois.

checks, in the form of questions, should be used to validate the first BRC estimate. For example, if there is more than one author, do their BRC estimates agree? Were students of previous (representing a cheaper labor cost) used? Does this product include a manual? If there is no manual, how much more funding is required to create one?

THE 5% SOLUTION

After two years of experimentation, many iterations, learning experience with actual licensed software, and prior comparisons with market products, it was found empirically that taking 5% of the modified replacement cost yielded a reasonable first estimate of a nonexclusive selling price. For example, if the BRC is estimated at \$100,000, the estimated selling price of the software would be 5% of this, or \$5,000. This represents the price an individual author, university, national laboratory, or software distributor might set for direct sales to the public. It is a one-time purchase price for a single user, nonexclusive, nontransferable sale, license, with no right to sublicense.

Although the 5% approach is empirical, it does an excellent job in providing a first estimate of a mar-

ket-competitive price. Where software is economically priced and market-competitive, fully operational, and fully documented, most companies would probably purchase it rather than try to invent around it. Moreover, a mere 20 sales are required to recover the modified replacement cost. In considering the possibility of an exclusive license, one would prepare summing up all the nonexclusive licenses that would be sold at the market price.

THE MARKET RULES

Quite often, competing software does not exist. The authors probably would not have written their software if they had been able to find an acceptable commercial package. However, in some cases, as in Internet or linear programming software, there may be very serious market competition. Such competition will always drive the software price down. Where market competition is strong, the selling price of the software must be reduced to fit into a niche of similar products. If competition exists, the market rules in setting prices. As a reality check, the price estimate obtained by using the 5% solution should always be compared against

market competitors.

CASE STUDIES

This approach, validated with sales, has been employed for a number of software packages that have been sold in these sales proceedings. Table 1, "Summary of Software Replacement Costs and Pricing," provides eight case studies, with the software functionality being categorized as: algorithm, database, object-oriented programming, education/training, and Internet-related software. (Additional data, consisting of more than 10 other cases, are not provided here.)

The software case number is noted in the first column. "Total Replacement Cost" is the total R&D cost required to develop the package, including the cost for all of the researchers working on the project at full salary, with benefits and overhead costs. "Modified Replacement Cost" includes the same costs as the preceding column, but it also incorporates the staff's expertise and lessons learned if the software were to be re-written today for a direct application.

The estimated selling price, 5% of the modified replacement cost, is also noted. "Landed Selling Price"

SUMMARY OF SOFTWARE REPLACEMENT COSTS AND PRICING

Case Number	Total Replacement Cost (R&D)	Modified Replacement Cost (Estimated Selling Price (5% of M.R.C.))	Landed Selling Price	Competing Commercial Products
1	\$1,500,000	\$750,000/27,000	\$37,500/1 ¹	None
2	\$600,000	\$300,000/10,000	\$15,000	\$1,000-100,000
3	\$42,000	\$21,000/1000	\$100	None
4	\$200,000	\$100,000/3,000	\$1,000	\$400-10,000
5	\$90,000	\$45,000/1000	\$450	\$200
6	\$2,000,000	\$1,000,000/30,000	\$50,000/1 ²	None
7	\$1,700,000	\$850,000/30,000	\$5,000	None
8	\$100,000	\$50,000/1,000	\$500 ³	\$500-1,000
9	\$1,000	\$500/1000	\$500 ⁴	None

1. This software will be commercialized for a yearly contract fee to be determined. It was estimated that the yearly savings from using the software would be about three to four times the single-time fee selling price.

2. This software may be commercialized in other products, and royalties may be negotiated as a percentage of the total package sales price.

3. Two leading competing products, which sold for \$1,000 to \$1,500, had limited features (but one product) compared to what is sold for \$50.

4. This software is under an extremely competitive.

5. It is an online or on-demand with another software package as an add-on. May have educational and other discounts.

6. This represents a single education/training computer-DBS module, of which there are 10 modules in the entire package. To obtain the total project cost, all data should be multiplied by a factor of 10, because there are 10 modules. Although this software fits no direct competitor yet, the market price for a single training CD is within the \$100-500 range.

Table 1

represent changes made to the 5% price on the basis of competing products, "forward-looking" pricing methods, and other considerations. "Competing Commercial Products" represents the cost of similar products on the market, if any.

The total R&D costs (i.e., total replacement costs) given in Table 1 are significantly higher than the modified replacement costs, by three to eight times. This is expected, because the total R&D cost includes tangibles and intangibles that an application-directed effort would not have. The initial selling price estimate, based on taking 5% of the modified replacement cost, is generally within plus or minus 15% of the actual selling price. Considering that all of these price estimates were made about three to 12 months before the software negotiations or sales had occurred, this agreement is good.

The last column in the table shows the selling price of comparable products now on the market. When comparable products are unavailable, it makes sense that an R&D organization would create software to meet its needs. This is the proverbial "technology push," with a technology looking for a market (as opposed to a market-driven situation, where a market is looking for a technology).

When comparable products do exist, we ask, "How similar are they?" and "What is the range of existing products?" Some of the product pricing speed indicates that, although the technology has entered the market, it is not broad enough or versatile enough to meet the general public's needs. This disparity arises because the low-end products are not very versatile, while the high-end products involve considerable tailoring to specific end-users.

Nevertheless, it is useful to determine the pricing of market products if possible. It is not necessary to

find an exact product match, but only to get a sense of an acceptable "going rate." There is great variability in product features, user-friendliness, technical support and maintenance, upgrades, etc., and price comparisons alone may not necessarily indicate the value of a product. The MRC/RC ratio seems to be in a range of about 5:1 for most of the data, but this ratio may serve as a cross-check on MRC (and the 5% solution) if other checks on the MRC are unavailable.

Two of the software packages, Cases one and six in Table 1, were estimated from one-time purchases and made into yearly renewable licenses (Journals). This occurred because the value of the software was estimated to have large yearly savings (at least three or four times the 5% solution estimated price, and in some cases significantly more), and because ongoing R&D could provide updates, that in turn would provide further incremental savings. The 5% solution still provided a good initial estimate of pricing even when yearly renewable licensing was considered.

Databases represent another class of software in which yearly renewable licenses should be considered. Yearly renewable fees seem more a marketing consideration when that software was first internally reviewed. However, additional study revealed potentially large incremental savings. Although yearly renewable licensing has its own problems (e.g., royalty collection, license maintenance, technical service), one should consider it as an alternative marketing approach for some software. As an aside, the 5% solution has also been applied to educational training videos or computer disks (CDs), as noted by case nine, and good agreement has occurred. However, educational software is a very competitive area, and in my experience, the MRC generally tends to be higher than noted in this case.

CONCLUSION

A proposed method of pricing shrink-wrapped software may be summarized as follows:

1. Determine the MRC and take 5% of it to estimate a list selling price. The MRC is defined as the development cost required to create the same software today, knowing what we now know.
2. Distribute the MRC estimate by making sure the software is commercially ready. Verify that the author's MRC estimate is reasonable; a minimal ratio, the MRC/RC ratio is in the 3-4% range, etc.
3. Because the market for existing products, if there are competitors, consider revising the software selling price to fit into the niche of comparable products.

The 5% approach was developed after more than two years of licensing and evaluating software. It should not be assumed to be a final rule; the process will certainly continue to evolve and be modified and reevaluated as more experience is gained. The majority of technology licenses are not "home runs" bringing in millions, but "average" licenses that get the technology into commerce and may be very important to the licensee. The modified replacement method was developed on the basis of these average licenses. Note that value and price are not always synonymous and that technical support, user-friendliness, multi-users users' manuals, etc. are also very important in setting a selling price.

REFERENCES

1. WITKIN, S. and BUCKLEY, T. "Users in Power License," in *Microcomputers: Users' Guide*, J. Gilman, Ed., Pricing Licensing of Technology, in *Microcomputers: Users' Guide*, J. Gilman, Ed., Wiley, 1982.
2. WITKIN, S. "Learning Technology: A Practical Guide to the Commercial Process," *IEEE Computer*, pp. 22-24.
3. WITKIN, S. "Microcomputers: Users' Guide," *IEEE Computer*, pp. 22-24.
4. WITKIN, S. "Learning Technology: A Practical Guide to the Commercial Process," *IEEE Computer*, pp. 22-24.
5. WITKIN, S. "Microcomputers: Users' Guide," *IEEE Computer*, pp. 22-24.