LIKE MANY WHO make their livings as writers, I have a long love affair with one particular word processing program: Microsoft Word. We go back nearly three decades. Typical of aging love affairs, ours is still affectionate (well, in one direction at least—I have no idea how it feels about me) but undeniably different. What’s changed over the years is my declining lust to update as new versions are announced. The result is that the revenue stream between me and Microsoft has slowed. Fewer updates are available, and I’m less likely to leap at the chance to download a new version.

What has happened between me and Microsoft is replicated many times over in the relationships between application program buyers and their vendors. The old marketing model—user buys software and then buys lots of regular updates as they become available—is now showing signs of its age.

The Creaky Old Model
Truth be told, the model’s been fairly creaky for years. Chances are that the last few updates of anything you’ve purchased haven’t been motivated by your lust for the “new bell and whistle content” (NB&WC) they offered. This is particularly true of applications that have been around for a while.

No, the reason you tend to update is more because you’ve updated your hardware or OS, and the old apps won’t run in the new environment. They

continued on p. 98
could have, of course, but the various vendors elected not to support two-way compatibility. This worked in everyone’s interest except yours: new hardware required a new OS, which required new apps. It worked the other way as well: new apps used lots more compute power, and so required new hardware and a new OS. The vendors were practicing a game of one hand washes the other with three hands involved. But the mutual hand-washing model has also begun to break down. Your reluctance to upgrade anything has increased to the extent that it might force other linked upgrades. Vendors have taken note of this and pretty much refuse to play the mutual hand-washing game.

With declining revenues, software vendors would dearly love to entice you into a new arrangement, one in which you rent your software and thus pay regularly for it. This is called the subscription model. But the problem with this approach is that it needed to start earlier, while you were still lusting for NB&WC. As it is, you might find yourself satisfied with the software you own and disinclined to switch to an expensive subscription for a few minor additions.

**Fresh Air**

While the old marketing models were in decline, something new was happening. I can illustrate it with two software applications that have become indispensable in the relatively new realm of building and maintaining ebooks: Sigil, an epub editor, and BlueGriffon, a WYSIWYG content editor for HTML and XHTML pages. (Because an epub is essentially a self-contained website, its code-level representation is HTML or XHTML.) Both of these enormously useful software apps are available for free.

As a marketing model, free presents an obvious problem. But as the Sigil and BlueGriffon websites make clear, there are ways to support the software makers. Because I use and appreciate both products, I send their developers money. I’ve done this repeatedly, trying each time to set the amount I send at the level of what I used to pay for my core software apps under the old buy-and-upgrade model. Maybe I’m unique in this, an old software developer who wants to make sure that developers of such elegant software are well rewarded. Maybe most people who use these apps don’t pay at all or pay less than I do. Turns out it doesn’t matter that much. While the revenue stream generated by Microsoft’s offerings has to support an organization of more than 100,000 employees, the revenue stream that Sigil generates all goes to one guy in a small town in Germany. This is the change that’s happening to the software market. Software used to be marketed B-to-B (business-to-business) or B-to-C (business-to-consumer). The new model is more like C-to-B and C-to-C. Important players in the software application market are tending less to be corporate behemoths and more to be individuals. Three things enable this:

- **Niche interconnectivity.** In our hyper-connected world, niches of people who do the same sorts of things are in constant touch with each other. If you build ebooks, for example, you quickly stumble on forums like mobileread.com. You lurk or pose a question, and you quickly learn what tools the other participants are using. It hardly matters that almost nobody on Earth has ever heard of Sigil, because all the people who make ebooks—the only ones who matter to Sigil’s maker—do know about it. Niche interconnectivity enables precision word-of-mouth marketing.

- **Brokerage.** If you’ve tended to think of Apple’s App Store and its lookalikes as nothing more than stores, you’ve missed their real import. The App Store is a brokerage, connecting software developers directly with the market. It has made possible an entirely new kind of startup: a pure development entity with no marketing, no collections, and no fulfillment. This is a dream come true for many would-be software developer/entrepreneurs: do what you do best, and let a brokerage take care of all the administrivia.

Our evolved marketplace facilitates a more direct relationship between software builders and buyers.
• Software reuse. The infrastructure now available to individual developers is so immense that it’s possible to create astounding products with relatively little effort. Consider the mobile app game called CandyCrush. If your company had been called on in the 1990s to bid for development of that product’s miraculous audio-visual interface, your bid certainly would have been in the tens to hundreds of millions of dollars. Today, so much of the miracle happens in the infrastructure that even a small team at a small company you never heard of can pull it off.

Each of these creates expanded opportunities for individuals and small groups to play in what was before essentially a corporate arena.

Our evolved marketplace facilitates a more direct relationship between software builders and buyers. C-to-C interactions have already become predominant in the space of software for end consumers. As the brokerage function matures, we’re likely to see brokerages handling not just goods offered but also goods needed. Companies will be able to compose their needed in-house applications through a series of brokered C-to-B mini-contracts for components. Software, which in the past was developed entirely inside corporate and organizational walls, will eventually be developed mostly on the outside, in homes and home offices of individual practitioners. Changing markets always create winners and losers. In this case, the winners are likely to be Davids and the losers Goliaths.

TOM DEMARCO is the author of thirteen books, most of them available as ebooks. His most recent is a novel, Andronescu’s Paradox. Contact him at tdemarco@systemsguild.com.