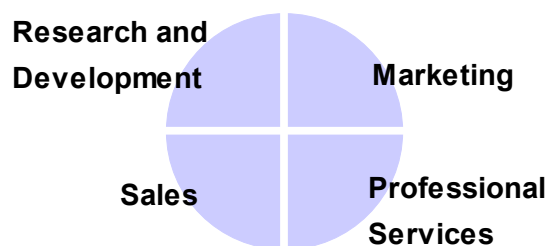# Cost Advantages of Open Source for a Profitable Software Business

This article on open source business strategies focuses specifically on the cost advantages of open source to the software publisher. Certainly, open source has cost advantages for end users. For example, this author is writing this article using OpenOffice, a free desktop package which includes a word processor, spreadsheet, database, and more. And for me, the primary benefit of OpenOffice is very clear: It's free. The software meets my requirements for word processing, without the cost of purchasing a commercial package such as Microsoft Word.

Just as open source provides cost advantages for end users, this license model also holds advantages for the software publisher. These advantages can impact various business functions within the organization, including research and development, marketing, professional services, and sales.

Figure 1: *Cost advantages of open source can positively impact multiple functions within a business.*

In some cases, the benefits are very concrete. In other cases, the benefits can be much less tangible. In fact, companies considering an open source model need to take a very close look at the reasons for choosing open source and then to validate the benefits that they will gain from this strategy. Frequently, these companies may recognize only marginal value for certain business functions (perhaps sales or development), which can potentially deflate the competitive advantage of an open source model.

In other words, the benefits of open source can vary dramatically for each company, and each organization should be careful to evaluate critically all of the advantages that they anticipate from this strategy. The software industry as a whole may at times appear to embrace certain benefits as universally applicable, yet these benefits may not in fact be appropriate to every company.

The summary below outlines some of the most common benefits that companies may gain from adopting an open source model. Any company considering open source should validate these benefits against the requirements of their own market segment, including customers, competitors, suppliers, and partners.

Advantages for Engineering Development

With an open source strategy, many companies effectively leverage their community of users to contribute enhancements, optimizations, and bug fixes to their code. Because the source code is freely available, any developer can download and modify/enhance the code as he sees fit. Open source companies frequently encourage their users to submit their modifications back to the company, for inclusion in a subsequent release of the product. The software publisher gains the benefit of a better product, and the end user frequently benefits by having those modifications supported by the

publishing company.

In addition, the open source users provide an excellent and easily identified audience for testing early releases of the product (what would be considered alpha and beta releases in most commercial software companies). Successful open source companies will typically foster an active community and an accelerated release cycle for both bug fixes and new features. As a result, the software publishers receive continual feedback regarding the stability and reliability of new software code, which in turn improves the quality of the software.

In fact, many open source software publishers adopt a quarterly or even monthly release cycle using Extreme Programming methodologies, compared to the 12-18 month monolithic release cycles of more traditional software companies. With a shorter release cycle, the open source companies can more efficiently collect feedback from the community about bugs and/or new requirements and implement the necessary modifications to meet customer needs.

Advantages for Marketing

Open source communities provide one of the most important and effective tools for marketing open source software. The community is truly the heart of the open source company. In the words of Jonathan Schwartz, CEO of Sun Microsystems, "Developers don't buy things, they join things."

Not only will developers join communities, but they will also contribute to the success of communities. These contributions, characterizing the evolution to Web 2.0, contribute heavily to the marketing benefits of open source by establishing a dynamic ecosystem around the open source company's technologies.

The user community provides an excellent means to publicize a company's software solution. Word of mouth within the community is one of the most effective ways to gain new users (both paying and

non-paying). I recently learned of a company that is moving to an open source strategy after having sold software exclusively under commercial licenses for several years. With the change to open source, the company was able to reduce the PR budget significantly because the marketing department could more effectively build awareness through the company's newly established community of open source users.

In addition to providing an effective way to build word of mouth about your product (including referrals), the community can correspondingly improve your web traffic (providing you have established a useful community presence on the web) as well as the number of prospects who may use your solution. For the company mentioned above, the shift to open source saw improvement in the company's web traffic by a factor of five.

Another cost advantage of open source lies in certain customer segments that mandate the use of open source technologies. In these segments where "traditional" commercial software companies cannot compete, marketing costs are significantly lower. The open source company can carry a much louder share of voice with the customer without having to match the marketing dollars spent by the traditional software publishers. In one example, the city of Munich in Germany decided to mandate open source software for the city government's desktop systems. This mandate excludes the incumbent, Microsoft, from competing. And consequently, the smaller open source companies avoid having to compete "dollar for dollar" with the Microsoft marketing behemoth.

Advantages for Technical Support

In many cases, software companies may rely on their community of open source users to support others in the community. For example, many open source companies use their communities to assist in any pre-sales technical support questions that users

may have. Consequently, they can effectively reduce their pre-sales support costs by relying on others who have downloaded the product for free.

Likewise, post-sales technical support issues may also be addressed through community forums or user groups, which can reduce the support staffing levels (and costs) for the software publisher. Of course, companies whose revenues rely on technical support (such as Red Hat) will consider an appropriate balance between community support and fee-based "commercial" support (the topic of a subsequent article). In most cases, the commercial support provides a much greater degree of technical knowledge and competence and is likely the most suitable alternative for issues that escalate beyond first level support. What's more, fee-based commercial support has the advantage of providing a guaranteed response that may not be available or feasible through the community contributors.

Because open source companies tend to have faster release cycles for their products, they may also experience less burden associated with patches and other issues tied to bugs in the software code. The software publisher can spend less time developing workarounds for earlier releases, because the company can make the bug fix available much sooner with a new, subsequent release. Of course, the success of this model depends on how quickly the customers will adopt new releases of the software that include the changes and fixes. By introducing the updated software more rapidly, the software publisher can help ensure that new downloads always provide the best quality code with the latest fixes.

Advantages for Sales

The advantages of open source to reduce sales costs are not as evident. On the one hand, sales can benefit significantly from the fact that the software is freely available for download and test by prospective customers. Consequently, the cost of software trials

are much lower. The sales representative does not need to work through the frequently time-consuming task of establishing a trial license along with the associated technical support costs.

At the same time, open source software publishers must invest in the necessary technology and programs to ensure the success of a technical evaluation that is unattended by the field sales organization. The success of a company's trial software program can significantly impact the cost of sales to move prospects through the technical evaluation phase of the sales cycle. Companies that do not develop the tools, documentation, and methodologies to provide a sufficient trial experience for their software must often make up for this deficiency through the sales force. The sales organization must frequently invest the time and effort to lead prospects through a satisfactory evaluation.

Another consideration for sales is the degree to which commercial (revenue generating) customers will develop "organically" from the open source user community. By building a thriving community, a software publisher will experience a certain conversion rate of open source users who develop into commercial users. This conversion rate can be very low (in most cases a small fraction of one percent of the open source community), and the resulting revenues are very unpredictable (both in terms of deal size and time to close). Consequently, open source companies frequently invest in sales programs to track and foster commercial opportunities. This investment provides much greater predictability for the business to forecast and manage revenues. I will address this topic in future articles discussing effective techniques for building and managing the sales pipeline in open source companies.

Conclusion

In sum, before adopting an open source business

model, companies should carefully weigh the specific advantages for their organizations. For instance, if you are developing a solution used by a very small community of users, then you may not receive a tremendous amount of feedback from the community when you release an early (beta) version of your product. So the value of the community does not necessarily translate to a cost-effective way to thoroughly beta test a software release, and the company will likely need to fund sufficient quality assurance testing in house to supplement testing performed by the community. The author encourages a common sense, methodical approach to adopting open source, including critically examining some of the "traditional" benefits espoused by the industry.

The author welcomes any comments on this article. Please send your feedback to jeff@jeffscroggin.com

Jeff Scroggin is a software sales and marketing executive with eighteen years of experience working with industry leaders such as Oracle, IBM, Hewlett-Packard, and EMC. He has held executive management positions at Oracle and Aspect Communications as well as early-stage startup companies. Jeff has extensive experience successfully marketing and selling software for commercial applications, bringing a unique combination of experience and scalability across the full sales and marketing lifecycle. Jeff earned his undergraduate degree in engineering at Stanford University where he was nominated a member of Phi Beta Kappa. Jeff received an MBA from the MIT Sloan School of Management.