Software Valuation Of Open Source Software (OSS) — It's Not All Free Beer!

By Dwight Olson

How do you put a value on free? That's the question facing investors in and business owners of open source software today. Few doubt that open source and open collaboration are reshaping the business landscape. Many believe there is only one open source landscape and it is "free beer" for everyone. However, those who believe that there is only one model called free beer, miss some very important aspects of value in the open source movement. It is very important to remember OSS (open source software) is not FOSS (free open source software) and they should not be confused with each other. For any information on FOSS, the reader may wish to surf the Internet with the key word "FOSS" and "free open source software."

In OSS when there is no direct revenue or income from the OSS product, valuations are undoubtedly a new challenge. Plus, current valuation methods and principles may or may not be the best method or principles to determine the value of any potential OSS investment transaction. Companies and individuals looking to have their OSS acquired or to be funded through venture investment must be familiar with current practices of valuation as well as possible new OSS methodologies and principles. It will be especially important for the OSS owner to be prepared to defend new asset value discussions and OSS monetization models.

One common OSS model is the corporate enterprise directed open source development. This is where a *corporate* America company (European or pick a region) invests in open source software development to achieve its business objective. One could suggest this business model is based on selling proprietary complements to free software. This OSS model is quite proprietary and any characterization as 'nonproprietary' is certainly surprising. For example, IBM uses this model to sell IBM hardware, ancillary software and service to the open source software and generates billions of dollars of revenue and is most assuredly proprietary.

Another common model is the "open-source-hybrid" model where companies have "tweaked" their business models to provide certain software free (including the source), but other software available only as part of a payment scheme. For example, a pc-based single user version may be "free beer" but access to the enterprise version is only available when you pay. This "open-source-hybrid" model also offers a similar and efficient model to the "proprietary" free trial before buying.

There are many reasons why an enterprise would opt for an open source model or collaboration and forego the possibility of making or investing in the old model of the proprietary licensing route. As mentioned earlier, some OSS companies gain profits in other ways than direct revenue license gains. Others suggest the reasons include cost savings, in terms of zero investment on seeking patent protection in various jurisdictions, productivity gains, as more and more people will be able to access and try the usage, brand building, since more and more people will become aware of the product in the markets and, most importantly, an expanded user base.

As the market expands, revenues from sales, one-off licenses, dual licensing, and complementary products and services may be enough to offset the opportunity cost of open source. IBM has used this approach very successfully. Today, many companies use some type of open source software, and many have plans to pilot it in the future. Utility and telecommunications firms, media companies, and public sector bodies lead enterprise adoption by a wide margin. European firms have been actively adopting open source solutions over the past four years. Today, almost 40 percent of companies already use some type of open source software. Forty-five percent of the firms using open source have deployed it in mission-critical environments, although the vast majority (70 percent) use it for non-key applications.[1]

Traditionally, copyright owners sold their copyrighted material in exchange for money. The lack of money changing hands in open source licensing should not be presumed to mean that there is no monetary or economic consideration. There are substantial benefits, including monetary benefits, to the creation and distribution of copyrighted works under public licenses that range far beyond traditional license

---

1. Is Open Source Gaining Adoption In Europe? Forrester, Manuel Ángel Méndez.

royalties. Even for an individual program creator who may generate market share by providing certain components free of charge. Similarly, a programmer may increase his/her national or international reputation by incubating open source projects.

The clear language of the Artistic License[2] creates conditions to protect the economic rights at issue in the granting of a public license. These conditions govern the rights to modify and distribute the computer programs and files included in the downloadable software package. The attribution and modification transparency requirements directly serve to drive traffic to the open source incubation page and to inform downstream users of the project, which is a significant economic goal of the copyright holder that the law will enforce. Through this controlled spread of information, the copyright holder gains creative collaborators to the open source project; by requiring that changes made by downstream users be visible to the copyright holder and others, the copyright holder learns about the uses for his software and gains others knowledge that can be used to advance future software releases.[3]

The terms of the Artistic License 1.0 were at issue in a 2007 federal district court decision in the U.S. which was criticized by some for suggesting that FOSS-like licenses could only be enforced through contract law rather than through copyright law, in contexts where contract damages would be difficult to establish. On appeal, a federal appellate court "determined that the terms of the Artistic License are enforceable copyright conditions."[4] The case was remanded to the District Court which did not apply the superior court's criteria (on the grounds that in the interim, the Supreme Court had changed the applicable law). Regarding legal OSS enforceability issues, as has been widely publicized in the industry, legal and even mainstream media, on August 13, 2008, the U.S. Court of Appeals for the Federal Circuit (CAFC) issued its decision in the closely watched case of *Jacobsen v. Katzer*. In its decision, the CAFC confirmed one of the core legal assumptions upon which the entire open source world is based—namely

that open source licenses are legally enforceable as licenses under U.S. copyright law.[5] Thus, free as well as open source has economic value.

OSS owners, investors, and the software valuation industry should be looking at alternative principles and methods for software valuation to augment existing mechanisms as we move into this new information economy. With the fact that OSS and collaboration software is a reality, and alternative principles and methods which might prove quite useful as ever-more-complex investment decisions in the OSS information economy are considered.

Valuation Using Past OSS Acquisitions

Augmenting any discussion of value with alternative principles may be a good idea, but, where do you start? Certainly, similar market acquisitions. If we look at some open source investment and acquisitions, Yahoo acquired *Zimbra*, an open source email and communications suite provider, for $350 million. Yet, Zimbra only received funding from Benchmark Partners, Redpoint Ventures and Accell Partners, raising $30.5 million with three rounds of investment funding. What prompted Yahoo to pay $350 million? Another acquisition was *BuzzTracker*, a tiny *news aggregation* site, for between $2 to $5 million. Then, Sun Microsystems became the owner of MySQL, the pioneering open source database system. Sun paid about $800 million in cash in exchange for all of MySQL stock and assumed another $200 million in options as part of the deal. Being aware of OSS acquisitions and transaction values may be a start, but with many believing OSS is another Dot Com bust, more than just market acquisition similarities may be required. Maybe the number of users of the open source software might prove worthy? Then maybe the OSS inventory, too?

Another Principle Might be to Look at the OSS Inventory

Software as well as open source software (OSS) is a complex bundle of assets and can consist of both Intellectual Property (IP) and Intellectual Asset (IA) components. One could consider the components

■ Dwight Olson,
V3Data, Principal,
San Diego, CA, USA
E-mail: dcolson@ ix.netcom.com

2. The Artistic License refers most commonly to a software license used for certain free and open source software packages. The name of the license is a reference to the concept of artistic license. See *http://en.wikipedia.org/wiki/Artistic_License.*

3. United States Court of Appeals for the Federal Circuit, 2008-1001, Robert Jacobsen, Plaintiff-Appellant, Matthew Katzer, Defendants-Appellees, case no. 06-CV-1905, Judge Jeffrey S. White.

4. ibid.

5. United States Court of Appeals for the Federal Circuit, 2008-1001, Robert Jacobsen, Plaintiff-Appellant, Matthew Katzer, Defendants-Appellees, case no. 06-CV-1905, Judge Jeffrey S. White.

of intellectual property (IP) to comprise patents, trademarks, copyrights and industrial designs because these four intellectual properties have legislation to govern, protect, and value propositions for the legal owners.[6] The other components or intellectual assets (IA) would then be considered to include domain names, trade secrets (non-provided source code) and know-how (other design digital assets), and the codified, tangible descriptions of specific knowledge which a software business uses to support commercialization (use). Software as well as OSS, by virtue of its component inventory[7] (or software inventory), can be the subject of various valuation strategies and discussions.

OSS Software Inventory

It is in OSS software commercialization (making it usable) where software takes on a software product shape (see Table 3). There is often confusion as to what software IP or IA to value and then how to value each type. For example, if there are patents or trademarks involved then the statutory rights granted can be used to value a given market monopoly (if one exists), to generate monopolistic income via a license program or to allow strategic value activities such as cross licensing, or to perform an investment IP management analysis.[8] Some believe that OSS should be valued only under patent or trademark. For those that do, are these the only value principles for OSS? What if there are no patents or trademarks? Does it mean that if there is no intent to maintain a monopoly, brand the product, or modify the GPL to include trademark restrictions, then there is no monetary value? Whoa! Maybe in some areas of technology, but certainly not open source software and open collobration. There is a big world of

open source and collaboration activities! For example, to date there are over 180,000 open source projects available in just the Sourceforge[9] alone.

In commercialization of the open source software product, the effort involved produces quantities of digital asset components. Some of the components will be provided under the OSS license and downloadable from the OSS forge (such as Sourceforge) and other digital assets will not be provided under the OSS, but retained by the company/individual for proprietary or other monetary purposes. The owner(s) will make the decision of which components are open and which are closed, see Table 1 and 2. Many software experts agree that in product commercialization, all project components need to be identified and governed, just as all other "hard" assets. No one would ever think of "not managing" inventory, buildings, or machines, yet for software some think a black hole is acceptable.

Software governance is asset-based and has value. Software development, as well as for OSS, produce component assets; each component comes into existence for a purpose. Someone determines that the component was/is needed to achieve "use" of the OSS software product just as for proprietary software. Sweat, in some cases paid sweat, went into their production. Of critical issue is whether these inventory component assets are development errors and not needed for use, or they are in fact digital assets in the sense that the stakeholders need to receive information concerning their use and ultimate value. If so, then the component has, as a minimum, a replacement value.

---

6. Intellectual Asset Identification, The First Step in an Intellectual Property Management Program, Dave Tyrrell and Gary Floyd, Vertex Intellectual Property Strategies Inc.

7. See Tables 1 and 2 below.

8. See *http://intellectualassetsinc.com/client-services/ma-analysis/*.

9. *SourceForge.net* is a source code repository and acts as a centralized location for software developers to control and manage open source software development. *SourceForge.net* is operated by Sourceforge, Inc. (formerly VA Software) and runs a version of the SourceForge software, forked from the last open-source version available. As of August 2008, SourceForge.net hosts more than 180,000 projects and more than 1.9 million registered users although it does contain many dormant or single-user projects.

Table 1. Owner Software OSS Components	
Trademarks	**Trade Secrets and Know How**
Marketing collateral	**Design documentation**
Patent(s)	**Other source code such as DRM**
Defensive if present	**Client databases**
OSS Copyrights	**Process know how**
Executable code	**Operating platforms**
User documentation	**Forge manufacturing instructions**
Source code	**Configuration data**
Build instructions	**QA test and procedures**
Domain Name(s) if no forge	**3rd party software**
Licenses	
Use and restrictions	

Table 2. OSS Software—Intangible Asset Inventory

Marketing and Sales	License Management
Marketing plans and collateral	DRM and license controls
Client Support Systems	Back office system
Installation and training	R&D Systems
User documentation and help	Internal design documentation
Client databases	Source code with comments
QA and Testing	Source code control with comments
Bug/support system	ASP databases
Testing code and data	QA test and procedures
Manufacturing System	3rd Party software
Specific build guides	Open Source & Strategy
Monetization Strategy	Client databases
Product plan (release and updates)	

Replacement Based Valuation for OSS Consideration

An OSS replacement valuation model must focus on the costs to reproduce the intellectual asset, not the historical costs. There is a value of the asset that is tied to the cost to recreate or acquire the *ownership* of the asset. Remember, possession is not ownership. In software asset valuations it is important to identify and value all software components, see Table 1 and 2. Appropriate costing models as well as historical or typical budgeting costs should be used with reasonableness equations to bias all results or typical budgeted costs.[10] Many modeling variables associated with developing software, such as the number of lines of new and modified code, were found to be useful in developing reasonableness equations.[11] A basic concern when considering any cost is its reasonableness. A statistical examination should be conducted, to develop a central value and an acceptable range of values for a software product replacement cost estimate.

In valuation of hardware IP, such as a computer hardware patent, one normally takes into account the physical depreciation and functional obsolescence of the hardware technology represented by the patent in calculating the replacement cost. At issue for software valuation as well as OSS is that software does not have physical depreciation as does the tangible hardware upon which it resides. Software should be looked at in terms of technology feasibility (which should include functional obsolesce) not depreciation. Methods used to depreciate tangibles as well as intangibles are based on the assumption that the goods being valued lose value over time. Such depreciation schedules are based on wear, the loss of value due to obsolescence, or changes in customer preferences.[12] However, well-maintained software in active use does not wear out, and is likely to gain value.[13]

Current thinking regarding software life cycle analysis is from the current accounting standard based upon the waterfall model first described by Winston Royce in 1970, ironically as an example of a flawed model for software development. The waterfall model is a sequential process, in which the life cycle is seen as flowing steadily downwards like a waterfall through the phases of Conception, Initiation, Analysis, Design, Construction, Testing, [Delivery], [Support], Maintenance[14] and then Obsolescence. Today, as probably for the past ten years, OSS software is almost always put into use with only partial functional requirements fulfilled, and then evolves over time; frequently over very long times as the software is enhanced, customized, adapted as new technical requirements and standards are discovered and changed.[15] One of the greatest inherent values of OSS software is its ability to evolve and adapt to new challenges through modifications, revisions, continuous improvement through open collaboration.

Important Physical Principles from VC's

Any business owner today developing software or a business unit that develops software including some

10. Software Valuation Part 2, *les Nouvelles*, December 2008.

11. R. W. Wolverton, "The Cost of Developing Large-Scale Software," *IEEE Trans. On Computers*, June 1974.

12. What is your software worth?, Gio Wiederhold, Communication of the ACM, September 2006.

13. Spolsky, J. Joel on Software. Apress, 2004.

14. Wikipedia reference for "waterfall model," *http://en.wikipedia.org/wiki/Waterfall_model.*

15. Software Valuation Part 2, *les Nouvelles*, December 2008.

aspect of OSS and expects to find equity or debt money for financing from any venture capital investor should be aware of the current ways they invest in a potential business, and this will most assuredly include OSS situations.

Venture capitalists evaluate their investment opportunities based on certain criteria. It is widely accepted that the three key investment decision criteria are management team, market projections and product.[16] In addition, venture capitalists have preferences—like venture's stage of development, its location, its industry or technology, and size of the investment required vary between one another. This criteria and preferences are related "to evaluation of an investment opportunity: does the venture have potential, is it worth our time and money, and does it fit our investment strategy?"[17] Venture capitalists base their evaluation on business plan, meetings with the entrepreneurial team, and various researches.

Valuation of high-tech companies by Venture Capitalists has been theoretically studied extensively. The value of a new venture is derived by discounting predicted future cash flows to the present.[18] The discounting factor depends on the probability of returns. Even if an enterprise has significant potential future cash flows, the risk of failure decreases its net present value. Different methodologies exist in the valuation, but all aim at answering the same question: what is the present value of expected future earnings or exit value of a company? The methods fall into four categories:

1) Liquidation value asset based methods,
2) Discounted cash flow based methods,
3) Options based valuation methods, and
4) Rule of thumb valuation methods (comparator valuations).

Companies and individuals looking for venture investment must be familiar with these current situation and valuation methodologies.

Undoubtedly for OSS VC valuations where there is no revenue or income from the OSS, then such methods may or may not be the best method to determine the value of the situation, but the business owner must be prepared to defend other asset value discussions. If not, then the business owner beware, the value will typically be based on *real property* and/or *real income*. A far cry for the current OSS information asset! ∎

Table 3. Software Product Components as Intellectual Property				
Software Component	Copyright	Trademarks	Patents	Trade Secrets
GUIs	√	√		
Source code	√	√	√	√
Object code	√			
Business processes	√	√	√	√
Data		√		√
Table structures	√			√
Documentation internal design and external use	√	√		√

---

16. Tyebjee & Bruno 1981, 1984; MacMillan, Siegel & Narasimha 1985.

17. ibid.

18. Frontiers of E-business research, 2005 Evaluation and Valuation of Open Source Software Companies: A Venture Capitalist Perspective.