Center for American Progress

1333 H Street, NW, 10th Floor
Washington, DC 20005
Tel: 202 682.1611 • Fax: 202 682.1867

www.americanprogress.org

An Open Source Tax Credit:

Proposal and Economic Analysis

John Irons and Carl Malamud*

Revised: March 13, 2006

Executive Summary

An open source tax credit is proposed which would allow individuals who develop open source software to receive a tax credit worth 20 percent of their out-of-pocket costs. Corporations and self-employed individuals may already take a deduction for their development expenses for both open source and proprietary commercial software. The open source tax credit provides a similar incentive for individuals who currently have no means to deduct these expenses.

Subsidizing open source software development can also be justified on grounds of economic efficiency. Open source software development enhances the ability of other developers to create new products. It also enhances the development

*John Irons, Ph.D., (jirons@americanprogress.org) is Director of Tax and Budget Policy and Carl Malamud (cmalamud@americanprogress.org) is Chief Technology Officer and Senior Fellow at the Center for American Progress (http://www.AmericanProgress.org/). The authors would like to thank John Podesta and Nat Torkington for their helpful reviews of initial drafts. Michael Powers provided research assistance.

and dissemination of knowledge and ideas more broadly. Since the benefits to the broader software development community and the economy as a whole go well beyond the users of an individual software product, a policy that subsidizes open source development would increase economic efficiency.

1 Introduction

In 2004 there were an estimated 1.2 million people who contributed to open source software in the U.S. One of the largest open source repositories, SourceForge.net, contains over 100,000 projects.[1] Open source software is an essential part of the operation of hundreds of millions of computers around the world and has become essential to the operation of the core Internet infrastructure and for many key applications, such as the World Wide Web.[2] Not only is open source software free to use and modify, it forms the basis of many commercial software products, e-commerce services, and many other for-profit ventures.[3]

Despite the growing importance of open source software development, the

[1] As of January 19, 2006, there were 110,678 registered projects according to SourceForge, and over 1 million registered users worldwide (http://www.sourceforge.net/, last accessed January 19, 2006). Over 20,000 projects are added per year, see S. Comino, F. Manenti, and M. Parisi, "From Planning to Mature: on the Determination of Open Source Take Off" Department of Economics, University of Trento, Italia; *Department of Economics Working Paper 0517*, 2005, available at http://econwpa.wustl.edu/eps/io/papers/0507/0507006.pdf, last accessed January 19, 2006.

[2] According to SourceForge.com, the top 500 projects have cumulatively been downloaded close to 1 billion times. According to Netcraft, an Internet survey organization, in January 2006 over 23 million or 66.40% of all active Internet web surveys were running the open source Apache software. See January 2006 Web Server Survey, Netcraft, available at http://news.netcraft.com/archives/2006/01/05/january_2006_web_server_survey.html, last accessed January 28, 2006.

[3] For example, IBM has spent over $1 billion in 2001 on such projects. See J. Lerner and J. Tirole, "The Economics of Technology Sharing: Open Source and Beyond," *Working Paper 10956,* National Bureau of Economic Research, November 2004, available at http://www.nber.org/papers/w10956, last accessed January 28, 2006.

tax code does not maintain balance between for-profit development and individual open source developers. For-profit companies can receive substantial tax benefits for costs associated with software development—both commercial software and open source software—such as the depreciation of capital equipment, expensing of other out-of-pocket costs, and may even receive a tax credit for R&D spending. Individual developers who devote their own time and resources to developing open source software, however, often receive no tax incentives for doing so.

In an effort to encourage open source development and dissemination, and to level the playing-field between individual and corporate development, we are proposing a 20 percent tax credit for qualified out-of-pocket expenses for open source software developers. Such a credit would have modest costs to the federal government, yet would add significant value to the U.S. economy.

The next section describes more precisely open source software and how it may be fundamentally different from for-profit products. The sections that follow describe the particular benefits of open source development and the imbalance between tax incentives for open source development between for-profit companies and individual developers. We then analyze the potential cost and economic impact of the proposed Open Source Tax Credit.

2 What is Open-Source Software?

Open source software is software that may be freely downloaded and used, and in which the basic operating instructions that make up the software are freely

viewable and may be modified by users. The software is produced across organizational boundaries, in many cases with corporate competitors cooperating in the creation of new solutions and, in many cases, the development costs being contributed without recompense by individuals. The creation of open source software has been much analyzed by economists attempting to understand what appears to be a new economic model for the production of resources. As Professor Yochai Benkler says in his seminal work, "Coase's Penguin, or, Linux and The Nature of the Firm":

> *At the heart of the economic engine of the world's most advanced economies, and in particular that of the United States, we are beginning to take notice of a hardy, persistent, and quite amazing phenomenon. A new model of production has taken root, one that should not be there, at least according to our most widely held beliefs about economic behavior. The intuitions of the late twentieth-century American resist the idea that thousands of volunteers could collaborate on a complex economic project. It certainly should not be that these volunteers will beat the largest and best financed business enterprises in the world at their own game. And yet, this is precisely what is happening in the software industry.*[4]

The ability to fully modify and customize software applications to suit one's purpose is unique to open source software. The nature of open software means

[4] Y. Benkler, "Coase's Penuin, or, Linux and The Nature of the Firm," *Yale Law Journal,* Vol. 112:2002, available at http://www.benkler.org/CoasesPenguin.html, last accessed January 30, 2006.

that a wide range of developers can build upon versions to speed the process of development, and can unleash the creativity of entire development communities-eventually leading to innovations that even for-profit companies can build upon and incorporate into their products.[5]

3 Open Source as Positive Externality

Standard economic theory describes the conditions under which a private market would reach an optimal outcome. One of those conditions is that the economic gains (or losses) associated with a private transaction are confined to those that engage in the transaction. In situations in which economic transactions have impacts external to those engaged in the transaction, the private market does not necessarily achieve an efficient outcome.

These effects are called *externalities* by economists, and can create efficiency-enhancing opportunities for government involvement. The textbook example of an externality is the case of pollution. Consider a company that creates a product for its customers, but that creates pollution in the process. If the pollution negatively impacts those living around the factories, then the product is said to create negative externalities. Pollution abating policy might then increase overall economic welfare. Positive externalities can also arise in economic transac-

[5]Indeed the creation of open source software has become critical to the continued development of high-end computing to the extent that a Presidential Advisory Commission found that support of open source software development by the Federal government "should be considered an important infrastructure investment." See President's Information Technology Advisory Committee, "Report to the President: Developing Open Source Software to Advance High End Computing," October, 2000, available at http://www.nitrd.gov/pubs/pitac/pres-oss-11sep00.pdf, last accessed January 28, 2006. Other researchers have examined the unique market for software and the relation between proprietary and open source products, see J. Lerner and J. Tirole, op. cit.

tions. Such is the case when a bee-keeper's hive helps to pollinate a neighboring farmer's field.

Externalities typically lead to a situation in which the costs (or benefits) of an economic transaction to individuals differ from the broader social costs (or benefits) to the broader community. In the case of positive externalities—in which others benefit from a private transaction—the private market will tend to under-provide the good, and government subsidies can increase overall economic efficiency.

Consider now the case of open source software. An individual developer can, if she chooses, keep her software code private and charge for that product. In that case the private gains would accumulate (in some proportion) to the developer and the purchaser (the end users). There are not necessarily any positive externalities in this transaction.

Suppose that the code were, instead, open source. The benefits from that piece of software would then accumulate to (a greater number of) end users *and* it would also benefit other developers. Users would also have the potential of even greater gain with open source, since they would have the ability to refine or enhance the product themselves.

Importantly, other developers benefit from the open-source code by (a) learning from the code, (b) incorporating the product or components into their work, and/or (c) by expanding/modifying the original code. Open source development can thus be thought of as disseminating knowledge and ideas in addition to providing a specific product to market. And as before, these benefits accumulate

to those who are not part of the initial transaction.

This positive spill-over of knowledge to a broader community creates an opportunity for public policy to improve upon the private outcome. By providing incentives for open-source development (either through R&D credits for private companies, or through tax credits for individual developers) the government would be increasing economic efficiency and potentially enhancing economic growth. [6]

By the very nature of open source software, an individual developer cannot capture all of the value generated by their efforts. The degree of openness of software leads to *both* additional social value *and* a decrease in the ability of developers to capture those economic gains—and thus to an inefficiently low level of open source development.

A special case arises when the software may generate (or be part of) a network. Network externalities arise when the value of the product (and access to the network) increases with the number of users of the network. In this case even proprietary software may create positive externalities. Open source software (and more broadly, open standards) however, may enhance the speed that a particular network is established.[7]

Since open source software is an essential part of modern information networks (especially the Internet), this would again argue for public support.

[6]The role of ideas in economic growth is heavily emphasized in the work of Paul Romer, see e.g. P. Romer "Increasing Returns and New Developments in the Theory of Growth." In *Equilibrium Theory and Applications: Proceedings of the 6th International Symposium in Economic Theory and Econometrics,* edited by W. Barnett et al., 1991.

[7]See C. Shapiro and H. Varian, *Information Rules: A Strategic Guide to the Network Economy,* Harvard Business School Press, 1999.

4 Playing Field Tilted Against Individuals

For-profit companies can deduct costs from pre-tax revenue for both proprietary and open-source products. Since these costs offset taxable revenue and can generate R&D credits, the after-tax cost of $\$X$ in development expenditures for a for-profit firm is $\$X \times (1 - t - c)$ where t is the tax rate and c is any qualified tax credit that the firm might be eligible for.[8] Also, individuals who have established for-profit ventures can deduct costs from their gross revenues.

Individuals who donate time and resources to open-source (and who do not have revenues from which to offset their costs) do not have this option.

A for-profit company may also receive a R&D tax credit in addition to the lower taxes from research expenditures. The Research and Experimentation Tax Credit, initially enacted in 1981, has been extended and modified throughout the last 25 years. The current version provides a 20 percent tax credit for qualified research and experimentation expenditures above a base amount. While the credit has expired as of January 1, an extension is currently pending in Congress. Also, small businesses and self-employed individuals can, through IRS Section 179 rules, expense the cost of most capital equipment.[9] Many states also have credits specifically designed for research as well.[10]

---

[8] Using a simple description of a profit-maximizing firm where after tax profits equal $(1 - t)[R(l, E) - wl - E]$, a marginal increase in expenditures would yield an additional $(1 - t)(\partial R/\partial E - 1) + c$, in after tax profits, where R is revenue, E is additional development expenditures, and c is the credit rate for additional development expenditures. If the firm were profit maximizing, the net impact of the expenditure on after tax profits would be zero in equilibrium. Ignoring the revenue impact, the marginal cost is $(1 - t - c)$.

[9] For further discussion of the legislative details of the R&D provisions, see Joint Committee on Taxation, "Description and Analysis of Certain Tax Provisions Expiring in 2005 and 2006," *JCX-12-05,* March 11, 2005, available at http://www.house.gov/jct/x-12-05.pdf, last accessed January 28, 2006, especially pp. 20-34.

[10] L. Paff, "State-Level R&D Tax Credits: A Firm-Level Analysis," *Topics in Economic*

The analysis of development of for-profit software is similar—additional resources spent on development can be subtracted from total revenue and can also trigger a R&D credit. In this case, because of the credit, a profit maximizing company would increase development expenditures beyond the point that would be justified by additional (non-credit) revenues alone. In either case, the true cost of increasing expenditure for development is less than the initial dollar outlay.

In contrast, for an individual who is developing open source software on their own time, the cost of expenditures on out-of-pocket development-related expenses is the full amount of the expenditure.[11]

Further, for-profit companies often have a greater ability than individuals to protect their work through patents and copyright.[12] The federal government, through the court system, provides significant protection for proprietary software.

Thus, the tax code, as currently written, creates a greater incentive for companies than for individuals to develop open source software. And the federal government helps to preserve the benefits of generating proprietary software. Individual developers, contributing software to our society for the common good

[11] The labor decision is more complicated. Say the hourly wage is \$100/hour. Spending that hour at one's job would pay $\$100(1 - t_i)$ after taxes, where t_i is the marginal tax rate faced by the developer. Devoting an additional hour to open source development costs $\$100(1 - t_i)$ in lost wages. For the for-profit firm, spending \$100 for an additional hour of developer's time would reduce profits by \$100 and would only cost the firm $\$100(1 - t_c)$ after taxes, where t_c is the appropriate corporate tax rate. The degree to which the for-profit and individual development are biased in one direction or another depends on the relative tax rates imposed on companies when compared to individual tax rates. Thus the argument for leveling the playing field is stronger for out-of-pocket expenses than for labor costs.

[12] While individuals can receive patent and copyright protection, the cost of enforcing those protections can be prohibitive.

and on their own time, are disadvantaged under this system.

5 Contribution to Public Good

In 2005, individuals received $36 billion worth of tax benefits as a result of charitable contributions.[13] Developers of open source software contribute both time and other resources to developing software code that is then added to the inventory of free software tools, as well as public knowledge more broadly. By providing developers with a credit for their out of pocket expenses, we are encouraging this kind of contribution to the public good in the same way as we encourage charitable giving.[14] In each case, the value of one's time is not included for tax purposes, but other expenses qualify.

6 Proposal: Open-Source Tax Credit

We propose establishing an Open Source Tax Credit for qualified out-of-pocket expenses of individual open source developers.[15] Such a credit would help to level the playing field between for-profit companies and individual developers. It would also encourage additional development of open source software and enhance dissemination of ideas across the economy.

[13] See "Fiscal Year 2007 Budget of the U.S. Government, Analytical Perspectives," January 2006, Table 19-1, p. 289.

[14] There is a difference in that the charitable giving incentive is in the form of a deduction, while the open-source incentive would be a credit. We feel that, more generally, incentives ought to be in the form of a credit so that mid- and lower-income individuals receive the same incentives as those with higher incomes.

[15] A tax deduction would be another vehicle, but we prefer a credit to ensure that the amount of the subsidy does not depend on the developer's income. A refundable credit provides the greatest incentive to those with lower incomes and is thus more progressive than a deduction.

Specifically, open source software would be treated like other individual tax deductions and credits. The value of an individual's donated time would not qualify—similar to the way charitable contributions are treated. However, out-of-pocket costs, such as fees for web hosting, the depreciated cost of capital expenses such as computers, travel to development-related conferences, and other expenses would qualify for a 20 percent refundable tax credit. We chose 20 percent as the amount of the credit after an examination of the literature surrounding the historical value of the Federal Research Tax credit.[16]

The credit is made refundable so that those with low incomes would still receive the incentive.[17] To limit the potential revenue loss from the open source tax credit, a limit on the total amount that an individual can claim should be considered.

As with any other deduction or expense, it would be the responsibility of the taxpayer to maintain documentation of the expenses and, if audited, to be able to demonstrate that the expenses are directly related to open source software development.[18]

[16]For example, see United States General Accounting Office, "Tax Policy and Administration: Review of Studies of the Effectiveness of the Research Tax Credit," *GAO/GGD-96-43,* May 1996, available at, http://www.gao.gov/archive/1996/gg96043.pdf, last accessed January 30, 2006.

[17]While we anticipate that most people engaging in open-source development will also have other sources of income, there may be cases in which developers might have temporarily low incomes. In this case, we should still want the tax code to preserve the incentives to undertake open source development.

[18]The IRS already has well-established rules for home offices, charitable contributions, and self-employment income which could be translated easily for an open source tax credit.

6.1 How much would it cost?

In 2004, there were over 1 million open source developers in the U.S. according to market research studies.[19] Of those, a significant portion may be engaged in for-profit ventures that would allow them to deduct their development-related expenses from their business-related revenue. We estimate that the fraction of individual, open source tax filers who do not have other means to deduct expenses to be 50 percent.[20]

There is no formal, systematic evidence on the amount of out-of-pocket expenses by individuals for open source development. Informal evidence suggests that half of all developers have no out-of-pocket costs; and for the remaining half, the expenses average $500.[21]

Assuming no behavioral changes to the tax code, an estimate of the revenue cost of the Open Source Tax Credit would be:

$$Total\ revenue\ cost = 0.2 \times N \times (\%\ individual\ filers) \times (expenses\ per\ person),$$

where N is the total number of open source developers. Given the assumptions above, the total revenue cost of the Open Source Tax Credit in 2004 would have

[19] North American Development Survey indicated 1.7 million people in 2004 contributed to open source in North America. The U.S. has 68 percent of the population in North America according to the CIA World Factbook, placing the number of developers at 1.16 million. Not all of these developers would claim the credit. See Evans Data Corporation, "North American Data Survey," October 2005, available at http://www.evansdata.com/survey_na_topical.shtml, last accessed January 30, 2006.

[20] C. Malamud, "Query to Internet Engineering Task Force Mailing List," Informal Survey, January 10, 2006.

[21] Ibid. A few developers seem to have significantly higher costs, and a cap on the credit may be appropriate in at least the first few years to contain costs, while conducting further research to more precisely determine the distribution of expenditures.

been $29 million.

This estimate will obviously change over time. In particular, the number of open source developers has grown significantly over the past few years, and further increases would increase the estimated annual cost over time.[22]

On the flip side, costs for computer equipment, hosting, storage, etc, continue to fall at a fairly dramatic pace. Continued technological progress that reduces out-of pocket development costs will tend to decrease the total federal revenue cost as well.

Encouraging open source development would, in many ways, save the federal government money. For example, NASA computers that were used to help send rovers to Mars were running on the Linux open source operating system. Additional (and higher-quality) open source packages would thus save the government money, which can be used for other purposes.[23]

Growth from companies building on open source software could possibly more than offset any possible decrease in tax revenue on corporate profits due to potential displacement of for-profit software by new open source software. For-profit companies who are able to harness open source software might also reduce their costs and thus increase their profits. Furthermore, if the credit were to effectively stimulate further innovation in the US software market, we could

[22]Simply projecting past rates of increases over time would be inappropriate since the past increases are not sustainable. From 2003-2004, for example, there was an estimated 54 percent increase in developers, see Evans Data Corporation.

[23]The California Department of Transportation, for example implemented an identity/password management project using open source software, saving the department an estimated $300,000. For more details, see State of California, "California Performance Review Report #4," available at http://cpr.ca.gov/report/cprrpt/issrec/stops/it/so10.htm#3t, last accessed Jan 19, 2006.

expect significant macroeconomic benefits. In either case, overall consumer welfare would increase as the amount of software in use increases and as prices decline.

More dynamic estimates of the costs of the Open Source Tax Credit would take into account any behavioral changes that the tax credit might induce. In particular, cost estimates would increase if the credit successfully encouraged more open source development, or encouraged greater development activity and associated out-of-pocket expenses.

To get a better sense of the magnitude of the cost or gain to the federal government in this case, if the 20 percent credit led to a 10 percent increase in both the number of developers and the qualified development expenses per person,[24] then the total revenue cost would be approximately $35 million. Larger behavioral effects would lead to greater costs, but also to more open source development (see Table 1.)

6.2 Economic Benefits

Generally speaking, many researchers have found tax incentives for R&D to have a statistically significant impact on real development activity. Most estimates seem to indicate that a tax incentive will lead to a dollar-for-dollar increase in expenditures, although there are also significantly higher estimates as well.[25]

[24] See the following section for more on the responsiveness of R&D spending to tax incentives.

[25] See B. Hall and J. Van Reenen, "How effective are fiscal incentives for R&D? A review of the evidence," *Research Policy* 29, 2000, pp. 449-469, for a broad summary of some of the empirical research. They conclude that "Taken as a whole there is substantial evidence that tax has an effect on R&D performed... A tax price elasticity of around unity is still a good ballpark figure." Although the authors noted that there is still significant variation in the literature. A more recent study of state-level (California) R&D tax incentives found a

Table 1: Responsiveness of revenue cost to number of developers and expenses: Annual estimate, 2004

Sensitivity (elasticity)	0	0.2	0.5	1	2
Developers (1,000s)	1,156	1,202	1,271	1,387	1,618
% Individual filers	50%	50%	50%	50%	50%
% filers with expenses	50%	50%	50%	50%	50%
Avg. Expenses	$500.00	$520.00	$550.00	$600.00	$700.00
Revenue Cost (mil)	$29	$31	$35	$42	$57

Table 1 assumes a 20 percent credit, hence $total\ revenue\ cost = 0.2 \times N \times (1 + e_n * 0.2) \times (\%\ individual\ filers) \times (qualified\ expenses\ per\ person)(1 + e_e * 0.2)$; where e_n is the elasticity of the number of developers with respect to the cost of out-of-pocket expenses; and e_e is the elasticity of the expense per person with respect to cost of out-of-pocket expenses. For simplicity, Table 1 assumes $e_e = e_n$.

To the extent that the Open Source Tax Credit increases the number and activity level of developers, we can expect (1) additional open source software products; (2) higher quality open source (and likely proprietary) products. Also, because of the nature of open source software, additional resources devoted to open source development will mean an increase in the quality of the current stock of existing software as developers will contribute more and better modifications to programs currently in use.

significantly higher responsiveness—with tax elasticities of over 20—see L. Paff, "State-Level R&D Tax Credits: A Firm-Level Analysis," *Topics in Economic Analysis & Policy,* Volume 5, Issue 1, 2005.

Quantifying the Economic Impact

To quantify the economic impact of the Open Source Tax Credit, a short example may be illustrative. A typical piece of consumer software can vary widely in price, but suppose that consumers value a piece of software, on-average, at $100. Obviously, there will be a wide range of valuations, with some people valuing the software more, and others less. And different kinds of software will of course be valued to different degrees as well.

The cost of producing open source software comes almost entirely from development costs—the marginal cost of distribution, once developed, can be very low.[26] If we assume the mid-point revenue estimate from Table 1 above (with a 10 percent increase in developers and costs resulting from the credit), we find that $35 million would be spent on the new tax credit, and would lead to an additional 115,600 developers, and an additional $30 million in qualified individual development spending.

If each new developer created just 1 new open source piece of software, and each piece of software were used by just 10 people on average, the Open Source Tax Credit would add $116 million in value to the economy.[27]

[26]The cost of distribution behaves as a step function, much like a public utility. Once you've borne the expense for a certain size server, for example, you can handle a lot of traffic. At some point, you have to step up to the next level. These costs, for things that are "big", can be quite hefty for an individual: perhaps $10,000 month. See also K. Schmidt and M Schnitzer, "Public Subsidies for Open Source? Some Economic Policy Issues of the Software Market," University of Munich, July 12, 2002, available at http://opensource.mit.edu/papers/schmidtschnitzer.pdf, last accessed January 30, 2006.

[27]A more precise estimate of the value of the economic impact would require detailed analysis of the impact of the credit on the number of developers; the number of hours spent developing software per developer; as well as the number and quality of projects. Data on these effects is unfortunately limited, but broader measures of the responsiveness of R&D expenditures to tax credits does show some sensitivity. See footnote 25.

If more than 10 people use the software on average, and if existing developers were able to increase the quantity and quality of their output, the value of the tax credit would obviously be much greater.

In either case, the benefit to the economy clearly outweighs the immediate revenue cost of enacting the Open Source Tax Credit.

While this example is illustrative, it is easy to see how even modest use of open source software can lead to large, economy-wide welfare gains. Also, this analysis glosses over the notion that open source software might be fundamentally different that software produced with an explicit profit motive in mind. As mentioned above, open source can be seen in many ways as a public good: since it is open and free to use, the software provides benefits to users and developers that proprietary software often cannot.

7 Conclusion

Much of the architecture that supports the Internet is built on open source software (Apache, Bind, Linux, and others) and by supporting open source, we also support the "network externalities" that are a hallmark of a connected information network.

As Shaprio and Varian point out, open systems can facilitate the development of programming skills and third-party, complementary programs; and hence spur overall economic development.[28]

[28]See H. Varian and C. Shapiro, "Linux Adoption in the Public Sector: An Economic Analysis" December 2003, available at http://www.sims.berkeley.edu/~ hal/Papers/2004/linux-adoption-in-the-public-sector.pdf, last accessed Jan. 25, 2006.

Given these spillovers, open source development should be encouraged since for-profit companies will tend to under-invest in these types of products. At the very least, the tax code ought not to discriminate against those that chose to develop open source software.