Understanding the Impact of 'Real World' Open-Source ROI

Published: 15 November 2011

Analyst(s): Mark Driver

Virtually all adopters expect significant ROI advantages from open-source technologies, yet most find that results vary from project to project. Here, we examine the foundation behind the open-source ROI proposition, and examine specific examples of successful open-source ROI scenarios.

Key Findings

- Open-source ROI advantages are driven by flexibility, innovation and cost optimization; all three work in confluence to establish long-term ROI differentiation over close-sourced alternatives. However, not every open-source software (OSS) project is created equal; instead, the depth (magnitude of differentiation) and breadth (number of enterprises affected) of OSS ROI impact varies widely from one project to another, and from one adopter to another.

- Open-source ROI differentiation is significant when adopters are fully able to exploit source code; in this scenario, typical ROI advantages will be significant over alternative approaches. The number of users able to fully exploit OSS ROI benefits at the source code level is typically a very small subset of the broader user community; however, this impact zone differs from one OSS project to another.

- Open-source ROI differentiation is moderate when adopters are unable to fully exploit the source code, and instead use the software as a "black box" (as-is and off-the-shelf).

- The number of users able to exploit OSS ROI benefits from black-box (as-is) usage can grow to a considerable size, depending on the domain and business value of the software.

- Open-source ROI differentiation is minor among adopters that leverage OSS technologies embedded in other IT products or services. The number of users able to exploit OSS ROI benefits from embedded usage can grow very large, depending on the domain and business value of the software.
Recommendations

- For all open-source investments, clearly determine where your interaction with the technology fits within a spectrum of ROI impact zones, ranging from: (1) white-box (source-level) use; (2) black-box (off-the-shelf) use; and (3) indirect (embedded) use.

- Approach open-source investments with a full-fledged ROI analysis that addresses not only potential cost savings, but also increased business value.

- Avoid setting expectations for one project based on the results of another; instead fully address the nuances of each engagement on its own merits; expect ROI to differ dramatically from one project to another — even within a single enterprise.

- Track investments over time to get a clear picture of real-world total cost of ownership (TCO) versus simplistic and incomplete total cost of acquisition (TCA) metrics.

Table of Contents

Analysis..................................................................................................................................................2
    What Is Software ROI, and How Do We Measure It?.................................................................2
    Case Study Example: Linux as Both a Success Story and a Cautionary Tale.........................3
    Factors That Contribute to OSS ROI Differentiation..............................................................4
    Zones Of Positive Open-Source ROI Impact............................................................................5
    How Can OSS Impact ROI Negatively?...................................................................................9

Recommended Reading.......................................................................................................................10

List of Figures

Figure 1. The Three Zones of OSS ROI Impact..............................................................................8
Figure 2. False Assumption of OSS ROI Impact.............................................................................9

Analysis

What Is Software ROI, and How Do We Measure It?

All software has an ROI that can be measured as business value minus cost divided by cost.

All enterprise software investments result in measurable ROI insofar as all software contributes to business value (even when it is zero or even a negative value), and all software also incurs costs, which is always greater than zero (see Note 1). While the equation for ROI is straightforward, accurately measuring it in specific scenarios can be an elusive task (for example, many software assets are routinely leveraged by multiple business centers within an enterprise). Moreover,
software assets often make segments within a larger stack, and contribute both directly and indirectly to business value in multiple scopes.

Moreover, discerning the true cost — TCO — of software versus the short-term TCA can be exceedingly difficult. The plausible measurement of business value that can be directly attributed to the use of a piece of software in the overall implementation of a business initiative is much more problematic. And, of course, if you can't measure business value, then you can't measure ROI (see "How to Link IT Metrics to Business Value" [Note: This document has been archived; some of its content may not reflect current conditions.]). Consequently, accurately measuring software ROI is as much of an art as a science. Nevertheless, while precise ROI mechanics can be elusive, Gartner's research shows that OSS can and does routinely impact software ROI in both positive and negative ways, compared with closed-source alternatives.

OSS solutions follow the same rules for ROI calculation that apply to all software; but the OSS model also introduces a number of additional variables that impact ROI in unique ways. Feedback from Gartner clients in real-world scenarios shows that OSS introduces ROI differentiation in some straightforward and obvious ways (e.g., cost of acquisition); however, in many cases, the OSS model introduces a set of more-nuanced ROI drivers in less obvious ways (e.g., shifting and obfuscating costs from capital expenses to operational expenses).

Finally, our research has shown that OSS ROI outcomes vary significantly from project to project (e.g., mature versus immature code), user to user (e.g., early adopter versus conservative user) and from one usage scenario to another (e.g., mission-critical versus situational). Here, we examine the key aspects of the OSS model that impact software ROI, and investigate how IT organizations can most accurately determine the true real-world OSS ROI potential in their own enterprises.

Case Study Example: Linux as Both a Success Story and a Cautionary Tale

Linux has been one of the outstanding successes of OSS due to a well-disciplined governance strategy and community participation based on merit, complemented by a channel of distributors that ensures the quality of code distribution. Linux has enabled Unix to be brought down to x86 commodity servers from the high-cost reduced instruction set computer (RISC) technologies, and to enable choice via subscription for support options.

However, most IT departments, have not analyzed Linux ROI, but rather its TCO; moreover, most have mistaken TCA for TCO. As the costs have escalated, due to more complex black boxes for clustering, grid, highly scalable, multisocket servers, and virtualization with unlimited guests, the subscription price has significantly increased.

Consequently, the more complex the stack and configurations supporting the Linux server, the greater the costs. In addition, users don't usually factor in the quality of service (QoS) role of distributors because they have adjusted to overall good quality. Consequently, when they have leveraged only a few incidences of service and support, users argue about why they should be paying subscription fees that appear relatively high when they are spread over hundreds to thousands of servers.
Today, many enterprises are questioning whether to employ community-supported distributions, rather than commercial ones. However, they don't know how to calculate their ROI, so they tend to become frustrated over the costs and wonder whether they should create free, unpaid versions in their infrastructures. This is becoming a common theme across many OSS technologies deployed in complex and mission-critical environments.

Factors That Contribute to OSS ROI Differentiation

To accurately ascertain if and how open source impacts ROI, compared with closed-source alternatives, we must first examine the unique characteristics that differentiate the model from the traditional closed-source software (CSS) approach. Most notably, the CSS model has evolved to maximize control and stewardship of software intellectual property (IP), so that IP owners are able to optimize potential commercial revenue streams. Toward this end, the CSS model is inherently skewed heavily toward a supply-side balance of control that favors the technology provider.

Alternatively, the OSS model promotes and enforces an open and collaborative approach to the development, distribution and support of software, wherein no single entity can exercise exclusive authoritative control over the software. This fact should not insinuate, however, that the open-source model is anti-commercial; far from it, numerous commercial strategies have emerged around the model in recent years (see the Recommended Reading section). The open-source model is, however, largely neutral regarding commercial business strategies; in other words, while OSS can coexist and even thrive with CSS, it is not defined by commercial success — that is, there is no inherent friction between the OSS model and the commercialization of products produced under that model.

At its heart, the OSS model aims to provide a universally accessible software commons — a supply of high-quality software assets that can be leveraged in whole or as a part of larger solutions by anyone for virtually any reason. The OSS model relies on four fundamental freedoms to accomplish this:

1. The freedom to operate (i.e., run) the software for any purpose
2. The freedom to study how the underlying program works, and to change it to make it do what you wish (i.e., examine and modify the source code)
3. The freedom to redistribute copies to others
4. The freedom to distribute copies of your own modified versions to others

The OSS model places its highest priority on the freedoms of the user, rather than the control of the IP owner — or any single entity in the software supply chain. As a result of this, the OSS model provides users and developers with a broad set of options that can potentially impact ROI in significant ways. For example:

- Users/developers can align software more directly to business value by customizing existing open-source code for their own specific needs (e.g., remEDIATE defects, add or change features, remove unneeded functionality, etc.).
- Users/developers can choose to support open-source solutions on their own, through a community effort or via commercial third-party vendors.

- Groups of like-minded users/developers can introduce and promote innovation in open-source efforts through community efforts driven by a network effect (see Note 2).

- Users/developers encounter lower barriers to entry that enable and promote their ability to experiment and prototype new ideas, without overcommitting budgets too early in the process.

- User/developers are never required to pay for the privilege of using open source; the TCA is driven virtually to $0 in most scenarios.

- Collaborative efforts around open-source development can potentially yield higher quality code and more secure code than closed-source alternatives.

Of course, this broad set of options doesn't guarantee optimal ROI results in every case. Instead, when the community dynamics around an OSS code base — technology users and suppliers alike — are best-able to exploit these options, the results can yield clear and significant ROI advantages. However, in other cases, they result in unrealized opportunities or, worse yet, illusions of positive ROI advantages that hide a negative ROI reality. For example, an overwhelming majority of OSS adopters report higher ROI results with open-source solutions, typically (but not exclusively) through reduced costs. However, in detailed discussions, only about 50% of Gartner clients are able to empirically demonstrate ROI advantages with OSS over sustained periods of time, compared with competing CSS.

Zones Of Positive Open-Source ROI Impact

Gartner's research (see the Recommended Reading section) tells us that OSS adoption is driven by three inter-related incentives: cost optimization, flexibility and innovation.

Recall that the formula for ROI is business value minus cost divided by cost. Flexibility and innovation are the principle mechanisms by which OSS drives business value differentiation, while cost optimization addresses TCO:

- **Flexibility:** Some OSS users can modify the code base to meet specific business needs, and they can mix open source into their own solutions to leverage high-quality reusable components. OSS users can also often leverage a competitive multivendor ecosystem for service and support; multiple vendors create competition, which drives prices down and avoids monopoly control over commercial support options.

- **Innovation:** Some users can jump-start their software development efforts with proven and well-established OSS assets, and improve upon them, avoiding the need to reinvent the wheel. Moreover, when developers share efforts and experiences, it establishes an OSS community that drives open and collaborative momentum; this creates a "network effect" (see Note 2) that dramatically increases the value of an open-source solution. As the community grows and promotes innovation, it draws on the efforts of and feedback from a wide developer/user base.
Cost optimization: To a large degree, cost optimization is itself a second-order effect of flexibility and innovation, but it can also stand on its own. Most obviously, open-source solutions dramatically reduce or even eliminate the cost of software acquisition. Moreover, reducing the barrier to entry for OSS solutions often expands the user base well beyond the scope of traditional CSS. This, in turn, fuels commoditization, consistency and interoperability, which can further optimize TCO over time.

Of course, not every OSS project is created equal. The depth (magnitude of differentiation) and breadth (number of enterprises affected) of OSS ROI impact varies widely from one project to another; however, a number of factors stand out:

- **White box versus black box:** Most software solutions are black-box assets insofar as they do not require access to source code for typical day-to-day use; however, some are white-box solutions that require an intimate understanding and/or extension (i.e., modification) of the underlying source code for typical usage scenarios (for example, many developer frameworks, even complex enterprise applications, require customization at the source code level to meet specific business and technical needs). White-box scenarios have the strongest synergy with open-source technologies and result in the highest ROI impact, compared with CSS alternatives.

- **Software complexity:** Software exists in a broad spectrum of complexity. Some solutions require highly trained elite developers, while most others benefit from the efforts of more-average journeyman developer ranks. Open-source solutions exist at all levels of complexity, but the solutions at minor to moderate complexity levels attract the broadest appeal (breadth of impact) across the developer ranks.

- **Business opportunity:** All software investments should be based on business value, but some solutions have a broader appeal than others. Some exist naturally within a small niche and are important to a relatively small group of community members, while others are more applicable to larger audiences. Open-source communities benefit greatly from large communities that drive critical mass, provide channels of new ideas, reduce potential bottlenecks of control, and, ultimately, drive maturity and sustainability over time.

- **Product maturity and governance:** Some software projects are well-governed and many more are not — this applies equally to open and closed source. However, the fully transparent and open nature of the OSS model makes it much harder for a project to hide in the mediocrity of obscurity. Consequently, poorly governed open-source solutions tend to die away naturally and quickly.

When considering maximum ROI differentiation, an ideal scenario would result from a well-governed OSS project with minor-to-moderate code complexity that is inherently a white-box solution with broad developer community appeal and a clear business opportunity. In other words, the ideal OSS scenario exists when a well-governed OSS can be leveraged in a situation where the client enterprise has the need and ability to alter the OSS to complete its business objectives.

Of course, no single variable governs the equation outright; instead, many attributes collectively drive ROI results. A number of factors can reduce theoretical OSS ROI potential, including: increasing software complexity, decreasing community size, immature code base, etc. At the
opposite end of the spectrum, a worst-case scenario for OSS ROI differentiation would result from a complex black-box-centric solution, with relatively small community appeal, an immature code base and limited business appeal.

In general, we see three distinctive “zones” of OSS ROI impact, these zones represent increasing potential impact scope, but decreasing magnitudes of ROI differentiation (see Figure 1):

- **OSS ROI Impact Zone 1**: Open-source ROI differentiation can be significant when adopters are fully able to exploit source code; in this scenario typical ROI advantages will be significant over alternative approaches. The number of users able to fully exploit OSS ROI benefits at the source code level is typically a very small subset of the broader user community. However, this impact zone differs from one OSS project to another. The size of this segment is often limited because: (1) there are relatively few enterprises that have both the right personnel (sufficiently skilled to be efficiently working on the OSS code) and the willingness to invest those personnel in this kind of work (the opportunity costs of using their best developers on the wrong projects are staggering); and (2) there are relatively few enterprise IT scenarios where the only way (or even the best way) to achieve the business goal is to customize OSS code.

- **OSS ROI Impact Zone 2**: The potential for OSS ROI does not end solely with developers able to directly leverage the four freedoms of the open-source model; instead, a second OSS ROI zone of impact is created when adopters leverage the broader OSS supply chain; the ROI benefits gained from direct ROI differentiation can be shared and passed on to community members downstream. OSS adopters who directly leverage the technology, but do not have the expertise to fully leverage the source code, fall into this zone of ROI impact. In these cases, enterprises often have a limited degree of internal technical skill allowing them to handle typical installation, configuration, patch management, etc.; however, they often do not have the acumen or the bandwidth to fully exploit open source at the source code level. In these cases, enterprises can still benefit from flexibility, innovation and cost optimization, even if it falls short of the best-case scenario. These users may still find unique business value from open source insofar as they:
  - Can decouple the software from a specific point of control, such as a specific vendor
  - Have the flexibility to manage certain levels of support themselves
  - Have the option of leveraging the community for support
  - Have the option of leveraging third-party vendors — ideally, from a competitive multivendor ecosystem
The size of this zone is generally much larger than Zone 1, but, as we can see in Figure 1, the impact of ROI differentiation is significantly lower as well. In these cases, much of the ROI value of an OSS project is tied to the robustness and effectiveness of the community supporting it.

**OSS ROI Impact Zone 3:** Interestingly, an OSS project can influence ROI results beyond its direct user base as well. A third zone of ROI impact includes IT organizations that leverage IT products or services that contain embedded OSS technologies. This zone is potentially much larger than the first two combined; but the depth (magnitude of differentiation) of ROI impact within this zone is much smaller. At this level, IT organizations enjoy a "trickle down" ROI effect created by other community members (perhaps vendors) that gain more substantial benefits from Zone 1 or Zone 2, and pass the value on downstream.
How Can OSS Impact ROI Negatively?

A significant majority of adopters routinely report positive ROI results from OSS adoption, yet Gartner research shows that only 50% of Gartner clients are able to empirically demonstrate these advantages (see Note 3). Of course, measuring software ROI is difficult and we can assume that some of these adopters enjoy benefits of open source regardless. But it’s also clear that conventional wisdom regarding OSS ROI impact is widely inaccurate; many assume the ROI impact across the three zones looks like the depiction in Figure 2.

Figure 2. False Assumption of OSS ROI Impact

![Figure 2: False Assumption of OSS ROI Impact](image)

Source: Gartner (November 2011)

The most common issues that arise concerning OSS ROI expectations versus realities are created by unrealistic goals. For example, an enterprise might extrapolate ROI expectations from one successful project to another, even when significant differences exist between them.

Confusion between TCA and long-term TCO is another common error. True TCO metrics must be established over time (typically three years or longer); setting expectations for long-term ROI based
on initial investments over short durations is unrealistic and most often extremely inaccurate. Moreover, incomplete ROI analysis is common when addressing open-source investments. Many proponents significantly overstate the advantages of open-source options while understating the risks. For example, adopters might underplay or entirely ignore risks associated with catastrophic failures — failing to apply QoS and SLA thresholds appropriate for mission-critical deployments. Ironically, in many cases, open-source investments are victims of their own success. Because OSS technology is often very high quality, it works exceptionally well out of the box. In these scenarios, an OSS solution may run well for months or even years before a catastrophic failure occurs. And when a problem does arise, the enterprise does not have a disaster recovery plan in place.

Recommended Reading

Some documents may not be available as part of your current Gartner subscription.

"Survey Analysis: Overview of Preferences and Practices in the Adoption and Usage of Open-Source Software"

"Hype Cycle for Open-Source Software, 2011"

"A CIO's Perspective on Open-Source Software"

"Critical Strategies to Manage Risk and Maximize Business Value of Open Source in the Enterprise"

"Examining the Relationship Between Open Source and Intellectual Property"

"Predicts 2011: Open-Source Software, the Power Behind the Throne"

"Toolkit: How to Cut Through the Hype Surrounding Open-Source Software"

Note 1 ROI and TCO

Specifically, ROI is the net present value of benefits costs over a given time period (for example, the benefits of using OSS software versus the costs over a five-year period, discounted to today's dollars). Otherwise, TCO as an element of ROI is meaningless.

Note 2 The Network Effect

Wikipedia defines the network effect as: "In economics and business, a network effect (also called network externality) is the effect that one user of a good or service has on the value of that product to other people. When network effect is present, the value of a product or service increases as more people use it."

Note 3 A Gartner Client Base Bias

Admittedly, Gartner's client base is skewed heavily toward mainstream and conservative IT users — we expect that ROI differentiation is more pronounced among leading-edge adopters. Nevertheless, for mainstream adopters, the long-term impact of OSS on software ROI is far from clear.
<table>
<thead>
<tr>
<th>Regional Headquarters</th>
<th>Japan Headquarters</th>
<th>Latin America Headquarters</th>
<th>Asia/Pacific Headquarters</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Corporate Headquarters</strong></td>
<td><strong>Gartner Japan Ltd.</strong></td>
<td><strong>Gartner do Brazil</strong></td>
<td><strong>Gartner Australasia Pty. Ltd.</strong></td>
</tr>
<tr>
<td>56 Top Gallant Road</td>
<td>Aobadai Hills, 6F</td>
<td>Av. das Nações Unidas, 12551</td>
<td>Level 9, 141 Walker Street</td>
</tr>
<tr>
<td>Stamford, CT 06902-7700</td>
<td>7-7, Aobadai, 4-chome</td>
<td>9º andar—World Trade Center</td>
<td>North Sydney</td>
</tr>
<tr>
<td>USA</td>
<td>Meguro-ku, Tokyo 153-0042</td>
<td>04578-903—São Paulo SP</td>
<td>New South Wales 2060</td>
</tr>
<tr>
<td>+1 203 964 0096</td>
<td>JAPAN</td>
<td>BRAZIL</td>
<td>AUSTRALIA</td>
</tr>
<tr>
<td></td>
<td>+81 3 3481 3670</td>
<td></td>
<td>+61 2 9459 4600</td>
</tr>
</tbody>
</table>

© 2011 Gartner, Inc. and/or its affiliates. All rights reserved. Gartner is a registered trademark of Gartner, Inc. or its affiliates. This publication may not be reproduced or distributed in any form without Gartner’s prior written permission. The information contained in this publication has been obtained from sources believed to be reliable. Gartner disclaims all warranties as to the accuracy, completeness or adequacy of such information and shall have no liability for errors, omissions or inadequacies in such information. This publication consists of the opinions of Gartner’s research organization and should not be construed as statements of fact. The opinions expressed herein are subject to change without notice. Although Gartner research may include a discussion of related legal issues, Gartner does not provide legal advice or services and its research should not be construed or used as such. Gartner is a public company, and its shareholders may include firms and funds that have financial interests in entities covered in Gartner research. Gartner’s Board of Directors may include senior managers of these firms or funds. Gartner research is produced independently by its research organization without input or influence from these firms, funds or their managers. For further information on the independence and integrity of Gartner research, see “Guiding Principles on Independence and Objectivity” on its website, http://www.gartner.com/technology/about/ombudsman/omb_guide2.jsp.