This is a **low resolution**, **black and white** version of the article you downloaded. To download the whole of Free Software Magazine in *high* resolution and color, please subscribe!

Subscriptions are free, and every subscriber receives our fantastic weekly newsletters — which are in fact fully edited articles about free software.

Please click here to subscribe:

`http://www.freesoftwaremagazine.com/subscribe`

Convincing management to approve free software

Tips to better advocacy

Maria Winslow

T he grassroots efforts of system administrators have brought Linux and other free software into the mainstream. To be an effective advocate for free software at work, you need to speak the language of management and convince them from their point of view. This article discusses how to present your case, why your audience makes all the difference, how to hook them with proof of cost savings, and reveals two secret weapons for your quest to promote free software.

This article explains why bashing Microsoft won't help you in your case, which migration recommendations will seem the most practical and feasible to management, and how to present those recommendations in terms that management will respond to. It talks about how your goals are different from those of management, and how to adjust your approach accordingly. Finally, it demystifies return on investment (ROI) and shows how you can put together simple calculations to back up your case.

The golden rule of advocacy

The most important thing to remember when you are making a case for free software is to be sensitive to your audience. Who are you talking to? How technical are they? What is most important to them? You must address their concerns. Most people consider it a courtesy if you tell them that you're researching different free software solutions and you'd like to take their concerns into account as you investigate the options.

Tip 1: don't bash Microsoft

Criticizing Microsoft can be counter-productive in your advocacy for free software. There are three important reasons for this.

The first and best reason is that you really don't have to. Jupiter Research conducted a survey in 2003 of IT professionals in the small- to mid-size business market, asking what free software they were using and why. The number one reason for migrating to Linux wasn't cost savings, security, or reliability. Perhaps surprisingly, the primary reason stated for migrating to free software was to get away from Microsoft. Therefore, you don't have to say anything negative; your audience may be thinking it anyway.

The second reason involves a little pop psychology. Consider this: when you denigrate Microsoft, you insult people who only have experience working with Microsoft. Either your audience or at least a percentage of their reports may be in this camp, so be careful of insulting the base of their professional knowledge.

The third reason is that you will seem more credible and professional if you stick to the positive and resist the urge to use a derogatory nickname.

Tip 2: be practical

Focus on recommendations that will seem the most practical and feasible to your audience. Your campaign should begin on the edges of the enterprise, with systems that are less

critical. It's best to start with migration recommendations for systems that are less likely to cause unacceptable disruption. Remember to consider management's perception of disruption as you consider potential candidates. For example, e-mail systems can be a difficult place to start unless you have a compelling reason to promote a migration in this area. Management will naturally shy away from migrations perceived to have a high risk of disruption.

Basic internet infrastructure is usually the best place to start. Basic web servers, DNS servers, web content, spam/virus filters, and similar services are commonly trusted to Linux and free software. Sometimes it's possible to get approval for a Linux deployment on reused hardware. So don't overlook this possibility, especially for file servers. It's easier for management to say yes when the capital outlay is so low.

In your advocacy efforts, be sure to stress to management that free software adoption doesn't have to be radical; just a few systems in non-critical roles can add up to savings now and increased staff experience for future deployments.

Tip 3: lose the tech-speak

Sometimes it might seem that management doesn't take your technical advice seriously. Understand that your goals are different from management's goals. You want to work with the best technology, advance your career by keeping up with new technologies as they develop, and do your job efficiently using tools that will make your job easier. You're technology-centric. Management wants to hold down the fort, make sure nothing goes terribly wrong, try to keep people from complaining, and do it all within the budget. They are not technology-centric. Note that the actual technology doesn't make it into the list of most important goals—it's just a means.

The significance is that when you focus on the technology, management thinks you don't understand the big picture. To management, it's about making things work well enough to keep everyone reasonably happy within the constraints of a budget. If your pitch to management focuses primarily on the superiority of the technology, you're only giving them a small part of the story from their point of view. Even worse, if you argue about the technology with a colleague in front of management, you're likely to get even more pushback.

For example, you may recommend dumping the mail server for Sendmail on Linux. Your colleague may advocate Post-

fix because it's more secure. When you enter a discussion on the technical merits of each solution, your boss may tune out the tech-speak while perceiving that either solution may pose a risk by being the wrong choice. By focusing on the technical, and especially by debating technical points, you and your colleague introduced more risk (or at least perception of risk) into the equation. So lay off the tech-speak and focus on your two secret weapons instead.

Secret weapon 1: case studies

Management loves case studies. They are anecdotal, easy and quick to read, and easy to identify with. Most importantly, case studies feel like proof to management. Case studies usually reflect their peers—IT directors, mainly, and an endorsement from a peer goes a long way. So start googling.

As you search for case studies to help you in your recommendations, follow these two basic rules:

- Avoid case studies provided by vendors, as they usually sound too much like marketing materials to be truly credible. Your best source is always going to be a magazine because the analysis will be independent.
- Choose case studies that are as similar to your scenario as possible to help management recognize how your free software recommendation would work in your situation as well as it did in the case study. Consider whether the case study is reflective of your industry segment, or if the basic IT landscape is familiar.

To get started, look at (`http://www.windows-linux.com/articles`) a few of the case studies I presented in my book The Practical Manager's Guide to Open Source (`http://windows-linux.com/practicalOpenSource`).

Secret weapon 2: show them the money

The single most effective way to influence any decision to deploy free software is to prepare a cost justification. When you hand your boss a spreadsheet showing the cost savings gained, they will consider your migration recommendation more seriously. Fortunately, this doesn't have to be complicated.

The Practical Manager's Guide to Open Source

Return on investment (ROI)

Return on investment (ROI), also called the cost-benefit ratio, is the benefits gained divided by the costs to acquire the investment. You can think of ROI as "for every dollar spend, I get back X percent".

Benefit = the total amount of savings (proprietary price—free software price)

ROI = benefit / cost

To calculate the ROI of a particular migration, you need a few cost numbers. Whenever possible, use historical budgetary data for the best accuracy. Not everyone pays the same for software, so if you use the actual value your organization pays, your calculation will be more credible. You need to find out two figures: the cost of the status quo replacement system and the cost of deploying your free software alternative. Calculate the cost of the status quo system as everything your organization will pay if you don't go with free software. The cost of the free software deployment should not be $0. It's neither believable nor true. Be sure to include support contracts from a third party and outsourcing costs for deployment or staff training, even if you pay nothing for the actual software.

Why ROI and not TCO?

TCO stands for Total Cost of Ownership, which measures the cost of the system over its entire life. So why should you calculate ROI and not TCO? There is one simple reason to use ROI—it is easy to calculate now. ROI involves the costs and benefits of the acquisition of a system, which are known at the time of deployment. By its very nature, TCO involves guessing, simply because you can't know the total cost of the system until the end of the cycle. TCO also doesn't show the benefit gained by the organization, just the total cost over the life of the system. The TCO of paper and pencil is lower than desktop computers, but clearly most organizations employ desktop computers.

An ROI example

Take as an example a Windows to Linux file server migration. You're making the recommendation because the server, accessed by 100 users, is now five years old, and you're coming up on a scheduled hardware/software upgrade. This is the perfect time to make a migration recommendation.

To calculate the ROI of this recommendation, you need to know:

1. What is the cost to upgrade the hardware and software, if we remain status quo?
2. What is the cost to migrate to Linux and Samba?

Because eWeek Labs found Linux/Samba to be two-and-a-half times faster at file serving than Windows 2003 Server, you're also recommending that you reuse the hardware. Once you have the pricing, you can find the benefit to the organization of migrating (status quo cost - free software cost). Then divide the benefit by the cost to arrive at the ROI.

Status quo upgrade:

- Windows 2003 Server, Enterprise Edition, includes 25 client access licenses ($3999)
- 75 additional CALs ($2448.75)
- New server hardware ($2000)

Total cost of a status quo upgrade: $8447.75

Linux replacement:

- Debian GNU/Linux with Samba ($0)
- Initial installation support contract with local system integrator ($1500)
- Reuse server hardware ($0)

Total cost of a Linux/Samba replacement: $1500

Total benefit: 8447.75 - 1500 = $6947.75

ROI = total benefit (6947.75) / cost of free software deployment (1500) = **463%**

This example shows that a simple migration can bring a very healthy ROI, which will help convince management that free software is a good idea for your IT environment. Use this example as a base for your calculations, and be sure to get the most accurate cost information possible.

Making your case

Remember, the golden rule of advocacy is to know your audience and be prepared to address their concerns. Technology matters less to management than it does to you, so make an effort to tone down the bombardment of information and give them just the highlights. Recommend migrations that will seem the most feasible and least risky to management. Be professional and positive in your recommendations. Search for case studies that are relevant to your situation—they are one of your secret weapons. Additionally, it is worth taking the time to put together a simple ROI calculation. Using the example in this article as a base, you can impress management with your well-rounded view of free software in the enterprise and meet with better success in your internal advocacy.

Copyright information

About the author

As an Open Source Practice Leader with Virtuas, Winslow assists clients in understanding the technical and budgetary impact free software will have on their computing environments. Her recent book, "The Practical Manager's Guide to Open Source", guides IT directors and system administrators through the process of finding practical uses for free software that will integrate seamlessly into existing infrastructures, as well as understanding the costs and savings. She is a frequent speaker and author on the topic of free software. She can be reached via the Practical Open Source website (`http://windows-linux.com/about.html`).