Commercial Port Advocacy mini−HOWTO

Table of Contents

Commercial Port Advocacy mini−HOWTO

Doug Loss `dloss@seul.org`

v0.1, 29 December 1999

This document discusses methods that can be used to approach commercial software companies to convince them to port their programs to Linux.

9.Resources

1.Copyright Information

2.Why write this?

I read over all the other advocacy howtos for Linux that I could find (I've listed them in the resource section at the end). They were almost all addressed to convincing end users (either business or personal) that Linux could meet their needs on a day−to−day basis. That's a very useful thing to do, but it wasn't what I was looking for. Only the Linux Advocacy Project came close to what I was looking for, and it didn't quite cover what I wanted. I wanted something that would help me approach organizations making software for other platforms and convince them to port their works to Linux. Since I couldn't find any howto concerning that I decided to write one.

In this howto I'll cover how to approach software companies and what arguments may be most effective in convincing them to port their programs to Linux. I won't talk about trying to convince them to release their Linux ports under open source licenses. While that might be a good idea, I think these things should be done in small steps. Advocating a port to another OS is much more likely to meet with interest than advocating what the company may perceive as "giving away" product, or advocating a radical change in their basic development model.

3.Other efforts

There are other groups and individuals doing their parts in trying to get various software and hardware vendors to support Linux. Norman Jacobowitz is a consultant working with SSC, Inc. on an advocacy project that approaches the same problem I'm addressing here, although from a different direction. Here's what he told me about his efforts:

> SSC, Inc, publishers of Linux Journal, maintain a "software wish list" at <http://www.linuxresources.com/wish>. They are currently paying an outside consultant to use these results and other data to lobby marketing managers at ISVs to port their products to Linux. This is an effective, on going project to bring more native software to Linux; so please drop by the wish list and vote for your favorite software.

Andrew Mayhew has been trying to convince hardware vendors that making Linux drivers, or at least releasing the specifications so the Linux community can write the drivers, is a good idea. Here are his thoughts:

> I go to conferences. I was most recently at Networld−Interop in Atlanta (which was shortly followed by the Atlanta Linux Expo). There at Interop I went to many of the vendors with two agendas. First, I was there as a respresentative of the ISP that I work for and was looking for solutions. But secondly, I was there find out who currently had Linux support and if they didn't have Linux support, why they didn't and was the company considering it. It is interesting to note, that in the large Novell section, there were actually two Linux related sub−booths. Additionally, Cobalt Micro was there with their thin server, along with RedHat and Caldera. Fairly small showings in a nearly completely non−Unix related conference, but a showing none the less.

> Most of the companies that I was talking to are primarily hardware vendors. They already don't make any money off of their drivers. They just need to develop the drivers so that people will use their hardware. My typical approach to one of these companies was to first ask about the product in general, so that they could get through their marketing routine quickly, and then ask about driver support. When the only words out of their mouths would be Windows 95, 98, and NT, I would ask about other platforms explaining that I run in a multiplatform environment and would need interoperability between these platforms. In introducing the idea that they should support other platforms I would only slowly work in the idea of Linux as one of them. I found that introducing the idea that I wanted driver support for Linux right off typically got me a knee−jerk reaction which would basically have the person shutdown and try to find a way to get out of the conversation. But if you can get their defenses down then you can explain to them how, in general, all they would need to invest to get Linux drivers would be to openly publish the specifications for talking to whatever hardware and possibly providing hardware to key developers. The biggest argument to this, is typically, "We have some proprietary ways of doing X and don't want to have that information out in the open." The usual way around this problem is to explain that being able to talk and use a device does not normally mean having to know what proprietary tricks they are pulling. At least this fits with the wireless LAN and the Fibre Channel IP people that I

was dealing with.

One smaller company, which I think may attempt to find someone in the community to help develop a driver for them came up with an interesting solution around the proprietary issues as well. This being that they would have the initial developers sign NDAs for the hardware documentation, but the source code could be open source so long as the documentation in the source was not just a copy of what the company provided the developer.

For software companies, I think it is a very good idea to point out that there is nothing or very little available of their kind of software; whatever that area of software is. But it probably should also be noted what does exist. Of particular interest would be the development tools, the development support available, and possibly information about other porting projects. In terms of these other projects they would be interested in the porting problems that they have solved or would similarly be tackling.

4.How to choose companies to approach

First, you need to identify an area where Linux is deficient in programs. I wouldn't try to convince any company to port their web server to Linux. There are plenty of such programs available. I'd much rather use my time trying to convince companies to port games or kindergarten through 12th grade educational programs to Linux, as there's very little of those types available (none in the educational area, so far as I know).

Second, you need to identify companies that are most likely to be interested. I have no hard evidence of this, but I strongly suspect that your efforts will meet more success if you concentrate on second−tier companies rather than on market leaders in your chosen category. Companies with seemingly dated products or products that were aimed at obscure platforms would seem to be good targets. Their software may not be dated for Linux users since we're not always taken in by the hype factor. They may well be looking for new markets for their products as their present market dwindles along with their ability to compete. I doubt that Disney or Davidson and Co. would consider porting their educational programs nearly as quickly as say, Soleil Software or Topologika. Also, if a company provides its software for both Windows and Macintosh currently, it may be more likely to consider supporting yet another OS than one that is exclusive to the Windows or Mac world. If all the companies in your target category are Windows or Macintosh exclusive I'd try the Mac shops first, as a Linux port would give them a much greater market expansion (on a percentage basis) than it would a Windows−only company.

5.The art of cold contacting

What you will be doing is known in the fund−raising business as "cold contacting." This means that your "target" company won't have known that you'll be contacting it, and won't have been primed to hear your message. A 1% response rate is considered normal. You should be able to do better than that.

In a sense all target companies will be slightly primed to hear about Linux due to the remarkable amount of publicity it's been getting of late. In that respect your contact won't be completely cold. That's good.

The first thing to do is to identify someone in the company to contact. It's always best (if possible) to identify an actual individual rather than a job title. Depending on the size of the company and its organizational structure, your best bet is the head of program development. If that isn't a possibility, try for the head of whatever technical section the company may have. If *that* isn't possible, read over whatever bios might be available in the "about the company" section of the company's web site (they almost all have something like this) and pick the person who seems most likely to be intrigued by Linux and to become an internal advocate for a Linux port. Finally, if none of the above things works try contacting the head of the company. Incidentally, it won't hurt to contact more than one person in the company if your bio research shows somebody other than the head of program development as the most likely person to be interested.

Your initial contact should probably be via email. First, email usually goes directly to the person addressed rather than being filtered through various layers of the organization as postal mail and telephone calls are. Second, with email everyone starts equal. Physical presentation and elocution don't enter into the contact, so the logic of the message may be more apparent.

The subject of your message should be understated. "Make Millions Easily!" will just get your message deleted as spam. Try something like "A good new market for your programs," or "An overlooked market for your software."

The first sentence in your message should probably be a conditional apology for sending the message to the wrong person if the person receiving it is the wrong person. The next sentence should request that the message be forwarded to the right person and that that person's email address be sent back to you for future contacts. This has a few effects. The apology establishes that you're not a know−it−all and that you are polite. The request reinforces the politeness and quietly lets it be known that this won't be a one−time contact. That's important. It's a lot easier to blow off a message if you don't think you'll ever hear from the writer again.

That brings up another point. If things work out right, you won't be making just a one−time contact with this company. You will be signing up to be an outside contact for them, a source of information about things Linux. As such, there are some guidelines to follow in all your contacts. Be polite. Be patient. Be truthful. Be helpful. Stay apart from internal politics.

Be polite means responding civilly to all messages, even if you consider them insulting or moronic. Remember, "A soft answer turneth away wrath." Besides, it's just possible that you may have misunderstood the message. Asking for a restatement of the message to clear up its meaning can't hurt.

Be patient means answering what you consider obvious questions calmly and clearly, and answering them as many times as necessary. Email isn't real−time; you can take a jog around to block to cool down before answering yet another, "But doesn't a Linux port mean we'd be expected to give our products away?" message.

Be truthful means answering each question to the best of your ability, and saying "I don't know" when that's the correct answer. However, "I don't know" is only the first part of that answer; "but I'll find out and get back to you" is the rest of it.

Be helpful means going beyond just answering the immediate question and trying to address the reasons the question was asked. For example, one company asked me how it could publicize the existence of a Linux port if it did one. I mentioned the standard places (comp.os.linux.announce, Linux Weekly News, Freshmeat, Slashdot, Linux Journal). Then I brought the question up in the seul−dev−apps mailing list. The discussion there eventually started the development of the lu−news system.

Stay apart from internal politics means keeping a little distance between yourself and your company contact. However friendly your exchanges are, your role shouldn't be one of confidante but one of outside expert and advocate. You won't force the company into supporting Linux. You can only make sure they know about the opportunity and help them find the best way to take advantage of it.

You should probably be prepared to answer questions about why no one else in the target company's market niche is developing for Linux (if that's indeed the case), and what capabilities are available in Linux, such as multimedia. Are the available media players exploitable commercially? Do they run efficiently? You might also make the point that a port to Linux of a graphical program will mean a port to the X Window System and will mean that the program is much easier to port to any other OS that uses X, such as Solaris, AIX, or HPUX.

Andrew Mayhew brought up this point to me, and it makes a lot of sense:

> It has been my (albeit limited) experience that structuring the email/letter sent to companies with the idea that I or one of my clients is actually interested in their product at the beginning (or as closely as reasonable) of the document gets more results. Now, this is only really applicable is you mean it.

6.What to say

Okay, enough about what tone to adopt. What do you *say* to convince them that the Linux community is worth porting for? Here's what I did.

First, I explained as well as I could just who the Linux community is, and why Linux users would be a receptive group for my target company's products. I very carefully didn't exaggerate, and made sure that I explained when my figures were estimates rather than hard numbers. If someone comes up with a way of measuring usage, we may be able to get the hard numbers we need for market demographics, but till then we have to do our best.

Next I explained why I thought the Linux market would be good for the company to enter.

Then I laid out the ways in which the company's current program line could be ported to Linux. I must say that I lean toward using Abacus Research & Development Inc. (ARDI) Executor technology as a "wrapper" for Macintosh binaries as the easiest and quickest way to do that, but WINE and the TWIN library (as I understand it,these two groups are now working together) are all possible tools to help move programs to Linux without full−blown ports. Incidentally, I cleared my letter with ARDI before I mentioned any action that their engineers might be able to take for the company. You would lose credibility if the target company acted on your recommendation, contacted someone like ARDI, and was essentially told, "We don't know what you're talking about."

There's also Loki Software, which does ports of commercial software to Linux. So far they're done only games, but when I talked with the president of Loki a while ago he was quite willing to consider doing similar ports of other types of software.

Finally, I ended with a personal note on why I was trying to convince the company to port to Linux. Here's a copy of my standard letter; don't copy it word for word, but feel free to adapt its organization if you like:

7.My standard contact message

Dear Sir or Madam:

If I've sent this to the wrong address in your organization, I apologize; could you please forward it to the appropriate person? Also, could you let me know who that appropriate person is so I can direct future communications to him or her? Thank you.

I know that [insert company name here] has fine educational software programs for both the Macintosh and Windows PCs. However, I'd like to speak on behalf of a computer community that has heretofore been overlooked by the entire educational software industry; the Linux community. I'd also like to call your attention to porting tools that can make moving Windows and especially Macintosh software to Linux nearly trivial.

Linux is the fastest growing operating system in the world. It is impossible to put exact numbers on how quickly it is growing because it can be downloaded for free off of the internet. The last attempt to estimate the total user base in early 1998 came up with 7 million users world−wide in early 1998 with over 100% growth/year. The explosion of Linux publicity since then has undoubtedly kept that growth rate alive, putting the current market at around 20 million with rapid expansion for some time to come. That's more than the worldwide total of OS/2 users and is near (if not more than) the number of Macintosh users. In addition, these numbers should probably be in some part subtracted from the Windows and Macintosh user figures, as almost all of the computers Linux is used on initially had Windows or MacOS installed.

Just who are these Linux users? They are primarily male, technically educated, and in their early twenties. The first two traits are more standard than the last; Linux users range from the teens to probably the mid−40s in age. (I'm 46, so I had to extend the range at least that far.) This is a prime demographic for the educational software market. These are people who generally have or will soon move into well−paying jobs in technical fields, and who are often just starting families. Linux users are usually not very patient with MacOS or Windows, and so tend not to see or to consider software offerings for those operating systems.

To the best of my knowledge, there are currently *zero* educational software programs available for Linux. While many Linux users will write their own programs if they can't find anything to fit their needs, that has so far not been the case with educational software. This may be because Linux only originated in 1991, and has only experienced explosive growth in the last 2−3 years. The market is completely open.

This has only recently been noticed by the Linux community. As more of us have young children, the awareness of the need for educational software for Linux is growing. I'm sure it will only be a matter of time till someone begins to address this need.

What is the easiest, most cost−effective way to enter this market? It's certainly not impossible to start from source code and rewrite the operating−system−specific routines to work with Linux. That's what many Macintosh ISVs did when they wanted to enter the Windows market. However, there are easier ways.

There are "wrapper" programs available for both Windows and Macintosh programs, which enable them to run on Linux without having to be rewritten extensively. For Windows, there is the TWIN library from Willows Software <http://www.willows.com>. I don't have any direct experience with TWIN, but the Willows website gets quite specific on what Windows routines move across cleanly and what ones need some touch up. The WINE Project (<http://www.winehq.com>) also has Winelib, a similar set of libraries that Corel is planning to help develop and use in porting its office applications to Linux.

For Macintosh programs, Abacus Research & Development, Inc. (ARDI <http://www.ardi.com/>) has rewritten a substantial fraction of the Macintosh OS and toolbox routines, and makes this technology available in two different ways. Executor is available both as a Macintosh emulator for end−users and as a porting tool for Mac ISVs. Executor is available for Linux, DOS and Windows. A demo of Executor for Linux is included on the Red Hat 5.1 Linux distribution, the most popular commercial distribution. The engineers at ARDI are fluent in Macintosh and Linux and can evaluate how hard it would be to make a Linux version of your Macintosh software. In many cases it can be done without your needing to change a single line of code. A Linux version of your program created in this way can easily fit on the same CD−ROM as your Mac and Windows executables, thereby giving you a three−OS program on one SKU. The expenditure required to open up this potentially lucrative new market is relatively minor; certainly much lower than the cost of the ports many companies made in expanding from the Mac−only market into the Windows and Mac market.

Finally, a personal note. The reason I'm moved to write to you proposing that you enter the Linux educational software market is my 6−year−old son. I run Linux because, of all the operating system choices available, it best fills my needs and desires. If there were good educational software available for Linux, I'd snap it up. I know I'm not alone; I've heard from other parents every time I've mentioned the problem in various Linux forums. There's a market out here waiting to buy your product. Please don't disappoint us.

8.The final inspirational message

If you decide to take a turn at advocating commercial software ports to Linux, remember that in so doing you're not acting as merely one individual, but as a representative of our entire community. I know we're more honest than Microsoft; I like to think we're less self−satisfied than Apple; I hope we're more generous in helping other users in need than any of the other user communities. Show those qualities to the companies you contact−−honesty, humility, and helpfulness−−and you stand a good chance of being successful.

9.Resources

Here is the list of other advocacy documents I looked through before I decided to write this one. While they didn't cover the aspect of Linux advocacy I was looking for, they are well worth your time to read through. They have a lot of good advice on how to be an effective ambassador to the non−Linuxen out there.

- Linux Advocacy Project
- Linux Advocacy mini−HOWTO
- Why Linux?
- Why Linux?
- Why use Linux?
- Politics
- [No Title]
- Advocating Linux
- Linux FUD Factor
- Linux Myth Dispeller
- Linuxmanship
- Four Phases of Linux Acceptance
- To use, or not to use?
- Linux Compared to Other OSs