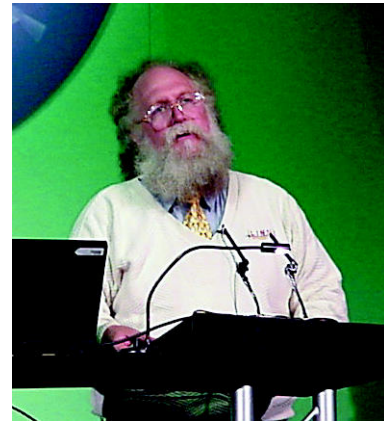


Trả lời phỏng vấn của Jon “maddog” Hall

1. Khi chuyển đổi sang PMNM, nên bắt đầu từ đâu, từ phía các máy chủ hay trên các máy trạm?

Thực ra các bạn có thể chuyển đổi sang dùng phần mềm nguồn mở - PMNM (FOSS) ngay cả khi bạn đang dùng hệ điều hành Windows. Có rất nhiều phần mềm nguồn mở hoạt động trên các hệ điều hành khác nhau. Chẳng hạn, chúng ta có thể dùng trình duyệt Firefox, phần mềm đọc thư Thunderbird, phần mềm OpenOffice cho ứng dụng văn phòng, MySQL cho cơ sở dữ liệu, và có nhiều phần mềm khác nữa. Tại địa chỉ OpenCD.org bạn có thể tìm thấy cả một đĩa CD với rất nhiều ứng dụng nguồn mở mà có thể sử dụng với môi trường hiện tại của bạn.



Sau đó thì tôi thường khuyên là nên bắt đầu từ phía máy chủ. Các quản trị mạng thì thường là dễ tiếp thu kiến thức mới hơn người dùng cuối cùng, hơn nữa số lượng quản trị mạng cũng ít hơn số người dùng. Nhiều việc triển khai PMNM tại phía máy chủ mà người dùng không nhận thấy. Sử dụng PMNM cho tường lửa, máy chủ DNS, máy chủ chia sẻ file và in ấn (dùng SAMBA) và các máy chủ cơ sở dữ liệu (sử dụng những thành phần sẵn sàng cao của PMNM) có thể được triển khai mà người dùng sẽ không nhận thấy, trừ việc họ cảm thấy thế nào về độ ổn định của các máy chủ khi chuyển sang PMNM. Các ứng dụng nền Web thường được cung cấp bởi Apache web server chạy trên nền một hệ điều hành Linux.

Bạn cũng không cần phải thay đổi các hệ cơ sở dữ liệu để có thể dùng một hệ điều hành nguồn mở. Oracle có thể chạy tốt trên hệ điều hành Linux. Trong khi người dùng đang làm quen với việc sử dụng các PMNM ở ngay trên nền các hệ thống họ đang có, thì các quản trị hệ thống được đào tạo và làm quen với việc giải quyết các vấn đề và trả lời những câu hỏi về PMNM.

Thêm nữa, không có lý do cụ thể nào khiến bạn không thử tìm kiếm PMNM cho các yêu cầu tính toán hiệu năng cao cũng như những yêu cầu của hệ thống nhúng của bạn. Các lập trình viên và người dùng của hệ thống hiệu năng cao thường khá phức tạp, và có thể đã sử dụng PMNM rồi. Người dùng các hệ thống nhúng không mấy khi nhìn thấy hệ điều hành, mà chỉ thấy các ứng

--

dụng. Do vậy thực tế là ngoài đặc tính miễn phí, có rất nhiều lý do để sử dụng PMNM cho các hệ thống nhúng.

2. Những khó khăn thường gặp phải khi chuyển đổi sang PMNM?

Không mấy khi tôi khuyên thực hiện một việc chuyển đổi người dùng từ một hệ thống này sang một hệ thống khác, trừ phi ta có một lý do về lợi ích thực sự, hoặc lý do về kinh tế để thực hiện chuyển đổi. Đừng thực hiện việc chuyển đổi nếu chỉ để nói rằng bạn muốn chuyển sang PMNM, trừ khi bạn chỉ rõ được lợi ích thực tế của việc làm đó. Tôi gọi đó là “all pain, no gain”. Ngay cả khi mọi việc diễn ra thuận lợi, người dùng vẫn sẽ bực bội với sự gián đoạn và những thay đổi trong công việc của họ. Tuy nhiên, sự thay đổi sắp tới sang Vista (trước đây hay gọi là Longhorn) cũng sẽ mang đến cho các bạn một cơ hội tốt để chuyển sang PMNM hơn là chuyển sang dùng hệ điều hành mới này của Microsoft.

Tôi khuyên nghị hãy xem xét thật kỹ lưỡng FOSS trong quá trình lên kế hoạch một dự án. Đừng liệt kê danh sách sử dụng những phần mềm như là “Microsoft SQL” hay “Microsoft Office”. Thay vào đó hãy liệt kê theo kiểu “phần mềm cơ sở dữ liệu quan hệ” và “phần mềm văn phòng gồm công cụ xử lý văn bản, bảng biểu và các khả năng trình diễn”. Bạn có thể tìm được câu trả lời giá thành rẻ hơn hoặc chất lượng hơn cho nhu cầu của mình.

Hãy nghĩ về việc sử dụng những dự án kiểu như dự án Linux Terminal Server (LTSP) để thiết lập máy chủ và một vài máy khách. Nó làm cho công việc cài đặt, quản trị, sao lưu và khôi phục hệ thống trở nên đơn giản. Mô hình lớp học, nhà băng, các điểm bán hàng, tất cả đều có thể sử dụng những dự án kiểu này với cấu hình cơ bản.

Một vài khó khăn khác mà bạn có thể gặp phải là sự không phù hợp về định dạng của các tài liệu. Tuy nhiên, bạn cũng đã có thể gặp vấn đề này ngay cả khi sử dụng giải pháp trả tiền như Microsoft Office mà phải chuyển từ phiên bản này sang phiên bản khác. Bạn cần xác định một Định dạng Dữ liệu Mở (Open Data Format) cho các tài liệu của mình, để đề phòng sự biến mất của một vài công ty và một trăm năm sau bạn không thể đọc được cái văn bản tạo ra từ trình soạn thảo do họ sản xuất. Bạn cần công ty đó có các định nghĩa định dạng dữ liệu một cách ĐẦY ĐỦ, thậm chí một vài đối thủ cạnh tranh có thể viết các sản phẩm tương tự dễ dàng đọc, viết loại định dạng này. Điều này sẽ giúp bạn đề phòng việc đánh mất khả năng đọc chính các tài liệu của mình, nhưng nó cũng sẽ cho phép các hệ thống chuẩn bị văn bản có thể cạnh tranh về tính dễ sử dụng và các tính năng khác mà thị trường chấp nhận.

3. Những quốc gia nào đi đầu trong việc tham gia và hỗ trợ PMNM?

--

Translated by Vu The Binh, Dang Quang Nguyen. NetNam Corp.

Tôi thích gọi Braxin là “Một ngôi sao sáng về PMNM”. Họ bắt đầu sử dụng PMNM vài năm trước đây chỉ với mục tiêu tiết kiệm. Nhưng sau đó họ có bắt đầu nhận ra rằng việc này đem lại nhiều việc làm cho đất nước họ thay vì phải đổ tiền ra nước ngoài để mua phần mềm. Họ cũng nhận ra rằng họ có thể gia công các phần mềm phù hợp với văn hóa của mình, và các phần mềm này sẽ là của họ chứ không thuộc về một vài công ty phần mềm. Điều này rất quan trọng, vì nó cho phép họ có thể sử dụng phần mềm cho bất kỳ mục đích gì mà họ muốn. Không ai nói với họ rằng họ không thể nhiều người cùng đăng nhập vào một “máy tính cá nhân” hay là họ không thể chuyển một phần nhỏ của phần mềm từ hệ thống này sang hệ thống khác (đó là điều mà các phần mềm thương phẩm thường quy định về bản quyền – ND).

Trong một thời gian dài, Braxin đã thực hiện tốt hầu hết các công việc trong lĩnh vực Công nghệ Thông tin bởi các doanh nghiệp tư nhân. Với PMNM, họ mở rộng công việc trong cộng đồng PMNM và tạo ra nhanh hơn các dự án, với giá thành rẻ hơn và có thể gia công các phần mềm có những chức năng phù hợp nhu cầu của mình. Tôi đã nhiều lần dùng Braxin làm ví dụ trong các bài nói chuyện của mình.

4. Những quốc gia nào đã có các chính sách cụ thể về PMNM từ phía chính phủ? Ông có thể cho vài ví dụ?

Có thể thấy nhiều nước đang thực hiện quá trình chuyển sang PMNM một cách rõ ràng. Malaysia có hẳn một chương trình quốc gia để thực hiện điều này. Nước Đức nổi tiếng với việc chính phủ đã chuyển các máy chủ, và tiếp theo là các máy trạm sang PMNM. Thành phố Munich và nhiều thành phố khác cũng đã thực hiện việc chuyển sang FOSS một cách công khai.

Hầu hết các nước sử dụng PMNM đều không thực hiện việc “chuyển đổi” để từ bỏ hẳn phần mềm mã nguồn đóng, nhưng họ đưa các chính sách là hoặc sử dụng PMNM cho các dự án mới, hoặc tính đến việc sử dụng song hành cả PMNM và các giải pháp độc quyền cho các dự án mới này. Do đó, vì nhu cầu mới mà các hệ thống cũ cần phải bị từ bỏ hoặc viết lại, PMNM sẽ thay thế các giải pháp độc quyền.

Nơi tôi thấy hay nhất để xem các ví dụ là:

<http://wiki.go-opensource.org/taskforce/FrontPage>

là Wiki được định hình nhờ chiến dịch “Tiến tới Mã nguồn Mở”. Thông thường do các đặc thù riêng tại từng nơi của chính phủ từng nước mà PMNM có các bước khởi đầu riêng, không nơi nào giống nơi nào. Malaysia lại là một ngoại lệ, họ có một chương trình quốc gia để chuyển đổi sang PMNM.

--

5. Những lĩnh vực nào sử dụng PMNM hiệu quả nhất hiện nay trên thế giới?

Chính phủ điện tử, chắc chắn thế! Giáo dục và các thư viện khoa học có vẻ cũng gắn chặt với PMNM, và còn có rất nhiều dự án trong dạng này.

Từ phía thương mại, ngân hàng và tài chính, chăm sóc sức khỏe, nghiên cứu y học (đặc biệt là nghiên cứu GENOME), truyền thông (VoIP và các hệ thống nhúng cho điện thoại di động), bán lẻ (đặc biệt là các hệ thống Điểm bán lẻ - Point-of-Sale), các công việc của hệ thống thông tin địa lý (đặc biệt trong việc định vị và tìm kiếm khí gas, dầu và các tài nguyên khác), công nghiệp điện ảnh (thực hiện các phim như “Titanic”, “Shrek”, “Lord of the Rings”) ... còn nhiều lĩnh vực khác nữa.

Trong thực tiễn, rất nhiều người đang dùng PMNM hôm nay mà họ không nhận ra. Sendmail được dùng để vận chuyển thư điện tử ngay từ những ngày đầu, và đó là PMNM. Apache là một web server PMNM, DNS server thường dựa trên BIND – một chương trình PMNM. Rất nhiều kỹ sư phát triển phần mềm sử dụng emacs làm trình soạn thảo văn bản, đó cũng là một trong những dự án PMNM đầu tiên.

Bản thân tôi dùng PMNM trong cuộc sống hàng ngày từ năm 1969.

6. Liệu có phải PMNM chỉ nên sử dụng ở những nơi không quan trọng, còn đối với những ứng dụng mang tính sống còn (Critical Application) thì không nên sử dụng PMNM. Điều này đúng hay sai và vì sao? Có thể nêu được các ứng dụng mang tính hệ thống sống còn dựa trên PMNM được sử dụng trong các ngành ngân hàng, hàng không, dầu khí, vũ trụ, quân sự hay không?

Không còn đặc tính cố hữu nào thuộc về phần mềm độc quyền mà làm nó an toàn hơn PMNM nữa. Nếu điều đó đúng, thì Microsoft đã không phải có nhiều vấn đề về an ninh đến như vậy. Mặt khác, một hệ thống an toàn là một hệ thống thường xuyên được cập nhật và khắc phục các lỗ hổng.

Tôi tin chắc rằng PMNM có hai đặc tính ưu việt hơn phần mềm độc quyền cả về an ninh lẫn độ ổn định cho các ứng dụng mang tính sống còn (critical application).

Thứ nhất, khi một hệ thống độc quyền ngừng hoạt động, bạn chỉ có một số giới hạn (ít) người có khả năng tìm và khắc phục trực trực. Chỉ có công ty đã làm ra sản phẩm đó mới có mã nguồn, vì cũng chỉ có một số lượng hạn chế các kỹ sư (có lẽ chỉ có một) trong công ty đó có hiểu biết về đoạn mã đó. Nếu kỹ sư đó đang đi nghỉ khi ứng dụng của bạn trực trực, thì có khi bạn phải đợi. So sánh điều này với PMNM, khi mà ai cũng có quyền biết mã nguồn, bạn có

--

thể thuê một kỹ sư độc lập để khắc phục sự cố cho bạn, điều này sẽ giảm thiểu thời gian gián đoạn, và rút ngắn thời gian khắc phục, sửa chữa.

Ưu điểm thứ hai đó là kiểm tra. Phần lớn các công ty mã độc quyền đưa ra bản “thử nghiệm” (thường gọi là Beta) cho một lượng khách hàng nhất định khi họ sẵn sàng để tung ra thị trường. Số lượng hạn chế khách hàng đó có thể, hoặc không thể đưa phần mềm đó vào hệ thống của họ để kiểm tra một cách thực sự. Đối với PMNM, mã nguồn được phát triển một cách công khai, và rất nhiều thực thể có thể tải về và thử nghiệm, đưa ra những phản hồi cho các nhà phát triển cho đến khi tung ra. Mô hình này có khuynh hướng xây dựng được mã nguồn ổn định hơn.

Mô hình Phát triển Mở (Open Development) cũng cho phép người dùng cuối đưa phản hồi trực tiếp và phát triển các tính năng mới như là “miếng vá” cho mã hiện thời, do đó các tính năng mới có thể được thử nghiệm bởi các hoàn cảnh thực tiễn trước khi nó được triển khai thực sự.

- 7. Làm thế nào để có thể làm việc tốt trong một môi trường hỗn hợp vừa Windows – based, vừa Linux – based, cả ở phía các máy chủ lẫn ở phía các máy trạm (nghĩa là trong một môi trường không triệt tiêu lẫn nhau giữa PMNM và phần mềm thương mại)?**

Như tôi đã trình bày ở phần trên, sử dụng các ứng dụng PMNM trên các hệ thống Windows là một trong những cách để chuyển đổi sang môi trường PMNM, và điều này vận hành rất tốt.

Linux có thể “mount”, đọc và viết vào các hệ thống file FAT, FAT-32, VFAT và NTFS. Với Samba, Linux còn vừa có thể là máy trạm, vừa là máy chủ cho các hệ thống Windows. Linux cũng có thể làm việc tốt với các hệ thống Apple, nói chuyện bằng giao thức Appletalk lẫn NFS.

Thông qua các máy chủ X Window System chạy trên Windows và thông qua các bộ trình duyệt (browsers), một người có thể thiết lập một “bàn làm việc – desktop” cho phép tất cả các ứng dụng có thể hiển thị trên đó, với các ứng dụng chia sẻ dữ liệu thông qua hệ thống file dựa trên mạng dưới đây.

- 8. It is currently having the trend of building the Single point of Authentication for all applications based on the LDAP system. However, we sometimes meet difficulties when integrating the AD (Active Directory) with OpenLDAP. Do you think that we can have a method to deal with this issue? particularly in the above-mentioned heterogeneous environment.**

--

9. Người ta đang nói tới một làn sóng phát triển các ứng dụng Chính phủ Điện tử (CPĐT) sử dụng PMNM. Điều này đúng hay sai và vì sao lại cần thiết như vậy?

Rõ ràng là có một xu hướng như vậy, và có rất nhiều nguyên nhân.

Thứ nhất, máy tính không còn là một cái gì xa xỉ để vận hành một quốc gia nữa. Không có quốc gia hiện đại không thể quay lại sử dụng giấy tờ và bút chì để quản lý nhà nước. Do đó máy tính, phần mềm trở thành một khía cạnh “nhiệm vụ sống còn” để vận hành đất nước. Khi phần mềm được nhập khẩu, và không có cách nào biết rõ mã nguồn trong các phần mềm đó, mà đang chạy cho các tàu thuyền, máy bay, xe tăng, tên lửa .v.v., chúng ta sẽ dễ gặp rủi ro. Khi không thể và lỗi hoặc cập nhật các chương trình cần thiết chỉ vì quốc gia bị cấm vận từ nước đã tạo ra phần mềm bạn đang dùng, bạn sẽ gặp rủi ro. Khi quốc gia đó có chính sách không cho phép bạn sử dụng các máy tính tốc độ cao và các phần mềm tiên tiến mà họ đang dùng, bạn đang gặp rủi ro. Làm thế nào để chúng ta có hòa bình trong một thế giới mà một quốc gia cảm thấy không ngớt rủi ro lòng thương các nước khác? Khi tất cả các quốc gia đều mạnh và an toàn, chúng ta sẽ bước một bước dài trong tiến trình hòa bình.

Ngay cả khi vấn đề an ninh quốc gia được gạt ra khỏi bức tranh, vẫn còn có vấn đề sự trường tồn. Việt Nam có một lịch sử hàng ngàn năm. Còn công ty phần mềm lâu đời nhất trên trái đất mới chỉ có 40 tuổi. Điều gì sẽ xảy ra khi một công ty mà Việt Nam mua phần mềm bị biến mất, hay nó được sát nhập với một công ty khác, và phần mềm Việt Nam đã mua không còn được sản xuất nữa? Liệu Việt Nam có phải chuyển đổi mọi phần mềm của chính phủ chỉ vì một công ty ở một quốc gia khác quyết định thay đổi nó? Gần đây đập thủy điện lớn nhất thế giới (ở Braxin) quyết định chuyển sang dùng PMNM vì họ nhận thấy các thiết bị của họ phải tồn tại ít nhất hàng trăm năm, và họ cần các phần mềm mà HỌ có thể bảo trì, nếu công ty cung cấp nó biến mất.

Cán cân thương mại là một vấn đề nữa. Rất nhiều quốc gia gặp rắc rối với vấn đề sử dụng trái phép phần mềm, và họ phải dừng việc đó lại để có thể gia nhập tổ chức Thương mại thế giới WTO. Thật không may mắn là điều này dẫn đến việc phải trả hàng triệu đô la cho Microsoft để có thể tiếp tục sử dụng các phần mềm họ đang dùng. Trong một số trường hợp số tiền này còn lớn hơn cả thu nhập quốc dân của quốc gia đó trogn một năm (GNP). PMNM cho phép các quốc gia đó có thể sử dụng các phần mềm trên mạng, gia công theo các nhu cầu bản địa (tạo ra việc làm và nâng cao trình độ bản địa) và không bị chảy máu ngoại tệ.

Niềm tự hào quốc gia là ý cuối cùng của tôi. Với các phần mềm bị sử dụng trái phép, thực sự không có sự phát triển của nền kinh tế cũng như hạ tầng phần mềm quốc gia. Con người phụ thuộc vào các phần mềm do người khác phát triển, và được sử dụng trái với mong muốn của họ. Nếu việc sử dụng trái phép này là hữu hình, như một chiếc xích lô chẳng hạn, thì sẽ bị gọi là “sự ăn cắp”. Làm thế nào để xã hội có thể tha thứ cho điều đó? Thậm chí ngay cả khi không có sự tranh cãi về đạo đức về việc sở hữu phần mềm, thì việc sử dụng trái phép cũng không thúc đẩy, khuyến khích việc lập trình. PNNM, cũng giống như nền kinh tế dịch vụ, mang cơ hội về các công việc trình độ cao cho những người muốn sống bằng dịch vụ thay đổi các phần mềm theo nhu cầu sử dụng. Con cái của các bạn sẽ thuộc về một thế giới của các lập trình viên, không phải như những người bị áp bức, mà đứng “bằng vai phải lứa” với họ.

10. Người ta nói mọi PMNM đều tuân thủ các chuẩn mở, điều này là đúng hay sai? Vậy các phần mềm thương mại có phải tuân thủ các chuẩn mở hay không? Làm sao để biết được điều này? Có thể giới thiệu một số các chuẩn mở cơ bản hiện nay là gì? Tham khảo được ở URL nào?

Rất nhiều các chuẩn được tạo ra xung quanh các dự án PMNM. POSIX được tạo thành từ Unix®, và cả System V® và Berkeley Software Definition(BSD). Các chuẩn Internet đang được Internet sử dụng ngày nay đều được tạo ra như là cá RFC mở (Request For Comments) bởi rất nhiều người quan tâm đến việc tạo thành Internet. Các chuẩn ngôn ngữ của IEEE và những chuẩn khác cho phép các dự án PMNM có thể tạo ra các bộ dịch tương thích (compilers). Chúng cố gắng phân đấu đi theo các chuẩn tự do, mở đã được công bố.

Khi các chuẩn không tồn tại, chúng ta tạo ra các PMNM mà không chỉ giúp ta thực hiện các chuẩn, mà còn giúp chúng ta tạo ra các chuẩn đó theo nhiều cách thức khác nhau. Bằng cách có mã PMNM, ngay cả các nhà cung cấp độc quyền cũng có thể hiểu rõ hơn những gì đã đề cập đến trong các chuẩn chính thức được viết ra. PMNM rất tốt cho việc phát triển và triển khai các chuẩn.

Một vài nhà sản xuất công bố rằng mã của họ là “chuẩn” chỉ vì nhiều người sử dụng nó. Nền công nghiệp gọi đó là “các chuẩn de-facto”, nhưng họ cũng không cam đoan với bạn rằng chuẩn đó sẽ không thay đổi đột ngột trong phiên bản tới. Họ cũng không mở truy cập cho mọi người để giúp xác định chuẩn nào sẽ tồn tại. Họ thậm chí cũng không tạo ra một sân chơi cho những người khác để thử triển khai “chuẩn” và tạo sự cạnh tranh cho người sở hữu các “chuẩn giả hiệu” (pseudo-standard).

Một vài nhà sản xuất thậm chí lấy các chuẩn đã được chấp nhận rộng rãi và thay đổi trong bản thân sản phẩm của họ, làm cho sản phẩm không còn tương

--

thích với các chuẩn nữa. Microsoft là một ví dụ, khi họ thay đổi cả Kerberos và JAVA thành các phiên bản không tương thích.

Để xem thêm về các chuyển biến của PMNM đối với chuẩn, hãy xem trang web nhóm các chuẩn tự do Free Standards Group (www.freestandards.org), là một địa điểm tạo các định nghĩa giao diện nhị phân cho tất cả các hệ thống PMNM.

11. Người ta nói không nên sử dụng PMNM nếu không muốn bị kiện về vi phạm bản quyền? Điều này là đúng hay sai và vì sao?

Độc quyền hay bản quyền phần mềm (software patents) là xấu xa và lỗi thời. Có thể đã có lý do tồn tại nó khi máy tính trị giá hàng tỷ tỷ đồng và số lượng các lập trình viên rất ít. Nhưng ngày nay bạn có thể mua một máy tính rất mạnh với giá vài triệu đồng, và phần lớn các học sinh trung học và sinh viên đại học đều biết cách lập trình và xử lý các vấn đề với những ý tưởng mới.

Sáng chế không phải là đạo luật của tạo hóa hay thượng đế. Đó là một phần của luật pháp được tạo ra để thúc đẩy xã hội và công nghệ. Sẽ tốt hơn cho chính phủ hay con người khi chia sẻ ý tưởng một cách tự do với hàng tỷ người có thể đóng góp các ý tưởng, hay tạo ra sự độc quyền được bảo vệ cho một số ít người? Tôi không nói rằng con người không nên có quyền cho ý tưởng của họ, nhưng có rất nhiều cách để bảo vệ cái quyền đó hơn là việc luật pháp đã tạo ra một phương thức kinh khủng được gọi là “bản quyền phần mềm – software patents”. Tôi sẽ chỉ ra rằng thời hoàng kim của các ý tưởng phần mềm thực sự đã xảy ra TRƯỚC KHI có khái niệm bản quyền phần mềm.

Phần mềm hiện nay ở khắp mọi nơi. Nó ở trong tất cả các thứ. Nó cũng giống như nghệ thuật, âm nhạc và toán học. Hãy tưởng tượng rằng bạn vẽ một bức tranh khi một ai đó sở hữu bản quyền về một nét bút vẽ, hay viết nhạc khi mà một ai đó sở hữu bản quyền về ý tưởng của một số nốt nhạc (“triplet” hay “riffle”). Và bây giờ bạn hãy tưởng tượng viết một chương trình với hơn 10.000 bản quyền áp dụng cho mấy thứ như “sử dụng con chuột” hay “làm tươi cửa sổ”.

Châu Á nên từ chối tôn kính độc quyền phần mềm (software patent).

Bản quyền (copyrights) lại là vấn đề khác. Khi một ai đó chỉnh sửa một đoạn mã, họ nên có quyền nói rằng cái gì đã xảy ra với nó. Tất cả các PMNM đều có bản quyền. Do đó, người sở hữu có quyền cấp phép (license) nó theo cách mà họ muốn, và thường nó được cấp phép theo rất nhiều cách, tùy thuộc vào nhu cầu của người dùng. Bản quyền (copyright) cho phép người ta thực hiện lại các ý tưởng theo các mã khác nhau.

--

Mã nguồn PMNM có thể tốt hơn, có thể tệ hơn, có thể an toàn hơn, cũng có thể kém an toàn hơn trong các vùng mà vi phạm bản quyền nhiều hơn mã độc quyền. Tuy nhiên, tôi có thể nói, rằng tất cả các lập trình viên PMNM mà tôi biết sẵn sàng “hy sinh” còn hơn là xâm phạm bản quyền của các lập trình viên khác. Và khi tất cả mã nguồn được mở để thẩm tra, rất ít khi người ta thấy một lập trình viên PMNM thực thụ lại sử dụng mã của người khác mà không được phép.

12. Nếu chúng tôi muốn thực hiện một dự án PMNM mới, ông khuyên chúng tôi thực hiện những bước nào?

Trước tiên, hãy nghiên cứu các dự án khác. Nếu đã có các công việc tương tự được thực hiện, hãy xem nó thành công hay thất bại. Đừng nên sáng chế ra những thứ người khác đã làm.

Một việc bạn cũng luôn nên cân nhắc đến, đó là thử thay đổi phần mềm của người khác và cố gắng chuyển lại cho người phát triển đầu tiên. Nó sẽ giúp cho những sự thay đổi đó nằm trong dòng chảy, và khi có các phiên bản mới, bạn không phải thay đổi lại lần nữa.

13. Làm thế nào để tránh xu hướng “Đóng hóa” PMNM? (Các công ty sử dụng PMNM để phát triển, sau đó đóng mã lại).

Khi các nhà phát triển quyết định đóng một dự án, GPL đảm bảo rằng những người dùng bản mã cuối cùng (theo GPL) có thể sử dụng nó mãi mãi, và cũng có thể dùng nó như là nền tảng cho dự án mới của riêng họ. Chúng ta gọi đó là “forking” một dự án. Nếu có đủ số lượng người tiếp tục dự án đã mở hoàn toàn, một dự án theo GPL, họ hoàn toàn có thể tiếp tục.

“Forking” xảy ra do cả các nguyên nhân khác. Đôi khi các nhà phát triển không nhất trí với hướng phát triển tương lai, và dự án bị chia thành hai, và cả hai vẫn theo GPL. Khi đó cộng đồng, thông qua cả các nhà phát triển và người sử dụng, quyết định nhánh nào sẽ tiếp tục phát triển. Như vậy, nó lặp lại sự cạnh tranh và lựa chọn, và đó không phải là một điều “tệ hại”.

14. Tương lai của việc ứng dụng và phát triển các PMNM trong tương lai như thế nào? Tương quan giữa PMNM và phần mềm thương mại trong tương lai sẽ là thế nào?

Mặc dù tôi tin rằng phần mềm nguồn đóng sẽ còn tồn tại trong thời gian dài sắp tới, tôi vẫn nghĩ rằng ý niệm về PMNM sẽ là phương thức phần mềm được tạo ra trong tương lai.

--

Phần mềm độc quyền ra đời vào đầu những năm 80 của thế kỷ trước, và thường được phát triển cho các môi trường nhỏ hơn và đồng nhất hơn so với nền kinh tế điện toán ngày nay. Vấn đề gặp phải của phần mềm độc quyền là nó không thể phù hợp nhu cầu của mọi người trên thế giới, và những công ty sản xuất ra nó không thể vừa đáp ứng nhu cầu đó, vừa vẫn có lãi. Điều này tước bỏ khả năng của người tiêu dùng cuối cùng (thậm chí người tiêu dùng là chính phủ) có thể nhận được các tính năng cũng như sửa lỗi mà họ cần.

Phần mềm nguồn mở, với sự sẵn có của mã nguồn, và sự tự do trong việc thay đổi mã, cho phép khách hàng cuối cùng, bất kể lớn hay nhỏ, có thể ra quyết định hoặc sử dụng phần mềm như nó vốn có, hoặc đợi đến khi nhà phát triển chính triển khai đúng cái họ cần, hay tiến lên và thuê một ai đó chỉnh sửa cho mình, theo đúng nhu cầu của mình.

Về người trả lời phỏng vấn

Jon “Maddog” Hall

Jon đã làm việc trong ngành công nghiệp máy tính từ năm 1969, sử dụng Unix từ năm 1977, và Linux từ 1994.

Ông đã từng là kỹ sư phần mềm, quản trị hệ thống, giám đốc sản phẩm, giám đốc marketing, và là giảng viên chuyên nghiệp. Jon trở thành Giám đốc điều hành của Linux International từ 1995, với bốn năm đầu tiên trong tư cách tình nguyện viên. Jon đã từng làm việc cho VA Linux systems, Compaq Computer Corporation trong nhóm Digital UNIX Marketing và Bell Laboratories và những công ty khác.

Trước đó ông là Trưởng phòng Khoa học máy tính ở Đại học Kỹ thuật Hartford, nơi những sinh viên yêu quý ông đã tặng cho ông biệt danh “maddog”. Maddog, như ông vẫn thích được gọi như vậy, có bằng thạc sĩ về Khoa học máy tính ở RPI (1977) và Cử nhân về Thương mại và Chế tạo ở Đại học Drexel (1973).

--

Translated by Vu The Binh, Dang Quang Nguyen. NetNam Corp.