
From: Jim Fredricksen
Sent: Friday, October 09, 1998 3:11 PM
To: Pascal Martin
Subject: FW: Much ado about Linux

thx for reminding, jim

Jim Fredricksen
Account Manager-OEM Sales
Microsoft Corporation
Tel: (425) 936-7268
Fax: (425) 936-7329 (bldg 18)
Jimfred@microsoft.com

-----Original Message-----

From: Raju Gulabani
Sent: Thursday, October 08, 1998 7:49 AM
To: Chameleon Core Team
Cc: Jawad Khaki; Mike Nash; John Frederiksen; Jennifer Cioffi; Mike Oldham; Jim Fredricksen
Subject: FW: Much ado about Linux

Please handle with care.

-----Original Message-----

From: Vinod Valloppillil (Exchange)
Sent: Wednesday, October 07, 1998 10:05 PM
To: Open Source Software/Linux (Private) (Exchange DL)
Subject: FW: Much ado about Linux

fyi... interesting reading. direct relevance to server appliance people

-----Original Message-----

From: Eric Rudder
Sent: Wednesday, October 07, 1998 9:42 PM
To: Vinod Valloppillil (Exchange), Oshoma Momoh; Oliver Sharp
Subject: FW: Much ado about Linux

just in case you guys haven't already gotten the fwd's.

-eric

-----Original Message-----

From: Nathan Myhrvold
Sent: Wednesday, October 07, 1998 9:23 PM
To: Bill Gates; Steve Ballmer (steveb), 'paulma', Eric Rudder; David Stutz; Jim Allchin (jimal), Rick Rashid (rashid)
Subject: Much ado about Linux

There has been a lot of interest in Linux as a competitor to Microsoft operating systems recently and I thought I would add some comments of my own to the issue.

A while back I wrote a memo on free software generally in which I took a fairly dim view of it. I won't repeat all of it here, but the punch line is that the only thing more expensive than commercial software with a license fee is "free" software. Having the software be "free" means that you cut out only one portion of the total cost - and as we all know from TCO studies, the license fee for software is only a tiny fraction. Free software needs to be maintained, tested and administered. That is where the real cost is. Eliminating the software developer only shifts the investment around. In particular, it is an inefficient shift because you take maintenance, testing and enhancement away from the developer (who can build expertise, economies of scale) and distribute it to a chaotic mix of smaller players, or even to end users. So from a basic economic viewpoint, "free" software is a fundamentally bad idea.

Only an idiot thinks that mass market software is expensive compared to any metric - cost of hardware, cost of end user time, opportunity cost... My God, even the cost of ELECTRICITY to run a PC for a few months to a year is higher than

Plaintiff's Exhibit
6482
Comes V. Microsoft

than the cost of the operating system! For a few hundred bucks we deliver what - a hundred million lines of code for Windows NT? Try developing that yourself for that price. Other commercial OS vendors do likewise. It's an infinitesimal cost from any perspective.

Diving into more detail, the "cathedral in the bazaar" theory is that the distributed approach to software development is better. The problem with this is that many of the key aspects of software development do not distribute well. Testing is the simplest example - if you have a volunteer army of folks writing their own extensions, who is going to test it all together? Testing, as we know from bitter experience, does not scale well. You need uniformity and control, which you don't get in a distributed, potluck environment where everybody hack their own features.

That covers broad theoretical aspects, but that isn't enough. Even if Linux is on a path that ultimately bumps against economic realities, it might take years, or even a decade for that to occur. In the meantime it could be an important competitor, wreaking havoc with established OS providers. There are several ways to look at Linux as a competitor.

As a desktop phenomenon, I don't think that Linux is very important. The application set is too limited, and they are too far behind. The place where Linux is very important (i.e. dangerous) is on the server.

This happens at an interesting time, because server based computing is exploding. The Internet creates a vast need for new servers at every level. The way I like to look at this is the ratio of CPU cycles (or RAM, disk whatever) on your local machine to the CPU cycles done on your behalf on servers - i.e. $R = (\text{Server CPU cycles}) / (\text{Client CPU cycles})$

In the early days of the PC business, $R = 0$ because there was no servers to speak of. These days $R = 1\%$ to 5% . This is because every PC is connected to servers for email, HTTP or some such, each of which typically has 100 or so users. The servers are more capable machines, but overall it still adds up to a few percent (try doing the math on your own usage pattern).

R is going to increase steadily, approaching 1. That is a VAST change in the industry.

The "thin client", "network computer" or "centralized computing" fans think that R will go to 1 by stealing functionality away from the client (i.e. doing word processing or other core PC applications on the server). This is wrong on several bases - instead it is new apps and expanded use of old apps (every citizen with an email address) that is, and will continue, to drive the penetration of new servers.

There are four significant aspects of Linux as a server OS:

1. Linux as the OS for server appliances - boxes which provide a very limited set of Internet connectivity, email etc for a set of users in a small business.
2. Linux as the new Netware - i.e. a simple network OS that provides a fairly limited set of services: file, print, SMTP, HTTP and so forth. In this mode application availability is not a big deal because you have a fairly limited set of
3. Linux as a host for Oracle or another SQL database.
4. Linux as a host for large scale custom server apps - i.e. Hotmail, Amazon.com etc.

Case 1 is the clearest, so I will treat this first. Simple, boring server apps are growing like crazy because everybody needs email, newsgroups and web pages. Small business users want something simple. I recently advised somebody setting up an office for a couple people. A turn key, appliance like server for basic connectivity needs would have made their lives a lot more simple.

We clearly need to compete like crazy in this space. Technically speaking this is where a stripped down embedded version of NT would be very useful - and as far as I know this is what we are doing.

Linux is not a particularly capable competitor in the sense that it has lots of special technology. Ideally a server appliance OS would have a lot of sophisticated self healing, remote admin and other features. To my knowledge, Linux is not at that level - it is being used because it is simple and fairly small. NT potentially has MUCH more to offer in this area. It also has some drawbacks (size, complexity, unneeded features). If we can strip out the unnecessary stuff, AND focus a lot of attention on the specific new technology for this area we could be very successful.

That said we really do have to worry about this area. But we need to temper the worry with some common sense. The whole point of a server appliance is to open up the market to users who could not afford to use a standard NT server - either for cost, or set up hassle etc. This expands the server market. It may also cannibalize some degree of current

NT sales However, it is not going to eliminate our server strategy altogether.

This is an old story. A new niche develops which is quite different from the industry mainstream. It catches on and grows like wild fire. Because it is a new area, the technology used to address it in the early stages is very simple, lean stuff. So, somebody at Microsoft panics and says OH MY GOD, THIS IS THE FUTURE, WE ARE SCREWED! They send impassioned email saying how everything we are doing is wrong because it is not like the dead simple stuff that is being used in the new niche.

There is some Linux mail like this in the last couple weeks. Prior to that there was some mail about how Amazon.com is the next killer app. This is healthy and there is a lot of value in this. The mistake that is made is to view the new area as a replacement threat, when in fact it is an incremental opportunity. A lot of bad decisions can be made when people confuse the two. It is great to be alert to changes in the market, but we need to recognize that most things are accretive, not direct replacements.

Another mistake is to think that the dead simple technology used to address the new niche is what we should be doing. This is tricky because the people in the new niche always talk a good game. It is easy to mistake their current state of technology with what is desirable, especially if they are ahead of us in that market. Although simplicity is sometimes a virtue, it is much better to focus on what the new niche NEEDS rather than how it is currently being addressed.

Returning to the case at hand, to beat Linux in the server appliance market, we need to add MORE technology, not less! Yes, we need to strip down NT to reduce resources needed to run it and other reasons, but at the same time we are stripping out irrelevant things, we had better be adding architectural support for features that will make a difference in that market. There is no point in competing with Linux on the basis of being a simple OS - that is fighting on their territory and their terms. If we tried to turn NT into Linux, they would be gaining in sophistication.

We need to address the incremental opportunity of server appliances, by getting somebody focused VERY hard on this area - both with embedded NT but also with thinking through what technology that market segment will need. My guess is that in the long run server appliances are like Web TV - a product for people who either would not have bought a PC, or who will graduate up to it eventually (in this case, for the server rather than client).

Case 2 is, to my mind, a broader threat than the server appliance. Novell IS the server appliance company - it started with proprietary 286 based servers for PCs, almost exactly like the server appliances. Later they switched to software only - but not a general OS - just software for simple network services - which at that time was mainly file and print service.

Fast forward to the present day - the Internet has given a new lease on life to simple server services. Instead of just file and print, the current generation of simple services are driven by the Internet - HTTP, NNTP, SMTP and various others. There is an opportunity for simple, high performance server software that runs on PC hardware and is a bit more flexible than what you can do with a server appliance. The market of strict server appliances will be smaller because most users will be unable to live with all the restrictions, but still want to use cheap PC hardware to accomplish their tasks.

If Novell had not self destructed, they should own this space, not Linux. Alternatively, it should be Novell and Linux duking it out for this market. Novell's almost complete abdication to Windows NT means that the simple server battle will be waged largely between NT and Linux.

Again, we need to gird for the threat. This is a challenge because our strategy against Novell was that we would have the flexible general purpose system, competing with their simple services. Just as we emerge victorious, simple servers come into vogue again - what a bummer!

The rise of the browser was a similar situation. PCs had rich documents with embedded graphics and images for years. All of a sudden the Internet browser was born and was a terribly retro, backward step technologically - it was about text with no choice of fonts, no structured graphics, only a couple image formats. All of the complex stuff that the PC industry had done seemed irrelevant, and the low tech stuff Netscape was using was paradoxically cool. Ultimately, the technology treadmill of improvements was what mattered. Netscape and Microsoft added feature after feature, and in the long run the better software development company won.

In the long run our flexible approach to servers with Win NT will win once again over Linux as it did over Novell. Linux will be unable to keep up with the pace of development. Once again, I want to reiterate that the way to compete here is NOT to try to adopt a Linux like stance - instead we need to add technology to improve our product for this market.

Case 3 is an interesting one. Large SQL databases have taken on so many low level functions that they are almost operating systems unto themselves. As a result, products like Oracle are able to support dozens of operating systems.

They can easily support Linux, and are likely to do so

Our strategy in this case is again to promote product features - such as cluster support, remote admin, ZAW and so forth to one up Linux. We have to assume that it will be a viable platform for Oracle and others, because they do so much of the work internally.

Case 4 is the topic of another round of hand wringing. Many web sites are basing their custom software on UNIX, and could easily do the same for Linux. Most current operating systems do not have much support for the kinds of problems these folks must tackle - nor for that matter to current email systems (designed for smaller scale...) or current databases (designed for fewer users, different usage patterns...). In the current early generation of systems like Amazon or Hotmail their creators have to do almost everything from scratch. In such a context, the OS does not matter very much - at least for now.

At the moment, Solaris and other commercial versions of UNIX are probably more of a competitive threat than Linux when it comes to large web sites like Hotmail or Amazon. In part this is due to better features, and in part because of Sun hardware. However, we should expect to see Linux used quite a bit in these contexts. However, looking forward my opinion once again is that technological innovation will provide us a way to make real improvements for this class of developer.

So, in summary I do not mean to dismiss Linux. It is a serious competitor which we have to counter with focused development and marketing activities. Unlike our usual competitors it has a unique economic model, without a centralized business behind it. In the long run this is a liability, but it can generate a lot of enthusiasm in the short run.

To counter the Linux threat, we need to focus development efforts on technological enhancements in the key areas that will matter to customers in the various segments - particularly case 1 & 2, but all of them should get some attention. If we do what we do best - creating and integrating new technology, we'll pull through OK.

Nathan