From: Nathan Myhrvold
Sent: Tuesday, October 01, 1996 12:05 PM
To: Bill Gates
Cc: Aaron Contorer
Subject: RE: Java runtime becomes the operating system

From: Bill Gates
Sent: Monday, September 30, 1996 9:36 PM
To: Nathan Myhrvold
Cc: Aaron Contorer
Subject: Java runtime becomes the operating system

I am worry a lot about how great Java/Javabeans and all the runtime work they are doing is and how much excitement this is generating. I am literally losing sleep over this issue since together with a move to more server based applications it seems like it could make it easy for people to do competitive operating systems.

I am very interested to get your thoughts on this. Prior to the advanced work you are driving what kind of defenses do we have against this? I certainly havent' come up with enough to relax about the situation and it is undermining my creativity.

I agree that this is a dilemma, but I don't think that it is severe enough that you should lose sleep over it. Here are some quick thoughts on the matter - I will put more time in on it also.

First, the excitement is overblown - at least from a *business* perspective. At the moment Java is expanding into a vacuum. It allows you to make cool web pages, and that is a very attractive thing for people. It gives programmers something new to learn, book people something new to sell books on, software tool companies a way to issue new development tools etc.

As you and know very well, this sort of widespread interest can become a self fullfilling phenomenon, because programmer attention creates programs. Some of these will be successful and that only fuels more participation in the phenomenon. However, at the same time you must keep something of a balanced perspective.

I think that the risk of Sun really taking the OS franchise away from us is much lower than the risk that they cheapen the entire business. They are so hell bent to give things away, and there is so much cross platform ferver that it will be hard for them or others to harness this energy toward a single platform In the limit, they can make the web totally OS agnostic - but there will still be other things that motivate one platform versus another.

In the very long run they could make it more and more difficult for us to keep up and thus even though the world is cross platform, we have more baggage, worse implementation and can't keep up. This is NOT going to happen quickly however - we will get several more swings at them

The new Java applications are NOT credible threats to traditional PC software any time soon. It is just insanity to think that they are. New things are NEVER a threat to the old world as soon as people say. Look at the mainframe vs PC. It has taken us TWO DECADES and even after all that IBM still has billions in mainframe revenue. Cool new technology always expands rapidly into NEW areas (where there is a vacuum). Pundits always say that this is going to kill the old businesses - eventually that happens but not anywhere near as soon as they say.

It is a new and uncomfortable feeling to be the incumbent rather than the challenger in one of these battles. However, we must not panic. The current perceptual battle is a long way distant from actual business and revenue issues, and we can't let the perceptual issues cloud our thinking too much.

This is not to say that Java is unimportant It is VERY important (just don't lose sleep!) I think that you are focusing on the wrong *kind* of threat. We are in danger of losing a new market which will grow at a pace which is very rapid indeed. This would be a tradegy to have happen to us, but it is different than a direct assult on our core asset, and our response must also be different.

The obvious things to do are:

Plaintiff's Exhibit
5803
Comes V. Microsoft

1 Provide our own means of dramatically improving web pages.

- Continue to "embrace and extend" - both at the level of new Java tools (like J++), and our broader browser strategy.

- Create some radical new approaches to improving web pages, or building web applications. I think that it is a big mistake to put all of our eggs in the "embrace and extend" basket. This thinking will lead us down the path to renounce any really interesting edge we could have. Over reliance on "embrace and extend" can lead to what I sometimes call the relentless drive to come in second, which does not help much in a winner take all world.

Note that by "radical" I mean more in concept than in technology - i.e. NOT speech or other really hard stuff. Java is pathetic technology. We do not need high tech here - we need some technology but mainly creativity in how we look at it.

2. Pioneer other means to participate in the new market. It is very rare that there is only one asset that matters. Hell, look at Netscape and Sun - each have an interesting asset, and this is still the EARLY stages of the net. There will be other technologies which matter and we should try to own one of them, even if it is in a totally different direction.

The key thing is to come up with something which will be buoyed up by the rising Internet tide Again, this need not be some super hard technical problem - it is more about being creative and finding an opportunity. Here are some examples (none perfect).

- Virtual worlds could be such a thing.

- New ways of doing server apps, or doing net transactions (a la Viper) could be such a thing.

I don't have as many examples here as I would like, but I am pretty sure that we could come up with some things if we really focussed on this area.

3. Continue to invest in our core business assets - in this case the Windows runtime. 99% of worldwide software revenue is still based on Windows apps, and way more than 100% of profits. If we let Windows decline in its non-web attributes then we will REALLY will be up shit creek. I fear that the overexcitement in the web, and the misunderstanding that our core business is only indirectly under attack means that our current course and speed neglects our best avenue to comete.

We need Windows to be the most compelling platform for users to choose Ideally this means that we win in every category. You are worried that we will only tie in the Java category because Javabeans and other runtime work will make cross platform really work well. I say we should try to tie (or win) with embrace and extend in the Java world. HOWEVER while doing this we MUST NOT allow Windows to lose in the other, non-web dimensions that are important to users selecting systems! There is more to a user's system choice than just the Web - and it is in those areas where we have a big advantage.

Put another way - suppose that Java is totally successful and totally cross platform - then why should people choose one system versus another? It will all hinge on other areas.

Here are some examples:

- We need to be the LEADER in multimedia - this means doing very cool audio and video. We want to lead with DVD and make it easy to . We must be the best platform for games (better than Ultra64 or Playstation). We want to be the best platform for graphics/video/audio authoring (better than SGI or Mac).

- Prevent barriers from coming up in ease of use. PCs must be instant on, they must go beyond plug and play to be even more self configuring, they must be self diagnosing. There is no technical reason why a PC shouldn't be the best possible user experience - better than those hypothetical web terminals. However, this takes WORK to make it happen It will take those web terminal guys some work too, but in our case there is a certain amount of baggage which will cause extra work for us, but will result in extra reward too. It is critical that we do this.

- Leverage the web to provide dramatic new functionality for old applications. The key example here is the "My Workspace" notion of totally abstracting storage on the net. This would be a dramatic change for users - storage abstractions are one of the hardest things to use in current PCs and we can make this all go away. If we do this based on STORAGE (i.e. file system APIs) it can short circuit the longer term move to distribution based on other APIs or protocols -- storage gets you most of the benefit.

I am frankly worried that we are not doing enough in this area because we have over-focused on the web We need end user focused people to really make our system great. We are handicapped by the fact that we also need to do an implementation technology shift (to NT) and a cultural/personnel shift (to Allchin's group which has historically focussed more on underlying technology than on end user stuff).

This is a big challenge to do these shifts AND keep the platform vital. The analogy with Apple is frightening. They pissed away their creativity on non-Mac things (Kaleida, Sweet Pea/Pippin, Taligent, Newton ...), and they tied up their Mac team with a technology shift (to Power PC). The analogy is unfair because the web work that Siverberg and Ludwig have been doing is much more relevant than what Apple did. However I still am worried that we are not making enough investment in our core asset. The good news is that we do have plenty of people we can focus on these problems, and if we do we can create some end user value that will be hard for anybody else to match.

I am not sure whether I managed to make you less worried or more worried. I think that this is a serious problem, but I think that we have more options than you may think - particularly with 2 and 3 above.

Nathan