From: Steven Sinofsky
Sent: Thursday, January 28, 1999 10:18 AM
To: Bill Gates; Jon DeVaan; Bob Muglia (Exchange)
Cc: Eric Rudder
Subject: RE: Leveraging Office

We want to talk about all of these. I was hoping for a small meeting on the 10th but this seems to have gotten a little larger and more formal which will make this more of a challenge for all of us.

I am concerned about planning the timing of the next release. We're working on this every day. I just don't want you to feel that there is a clear path to getting everything shortly. It is very complex to think about releasing any release of Office (customer churn, quality issues, expectations, etc.), and the shorter time frame the less we can do--and the less we do the fewer people will work on it (which means that we get less, and then have parallel development to deal with which risks us being out of sync with Platinum+ 1). This is only one perspective and I really just need you to be open minded because purly from a planning process we're not there yet on what a *platinum wave* means or when it is. We want to be, but it is much easier said than done -- especially to do it in a way that gets us in the Notes game in a measurable way and justifies the customer downsides of a release of office on the heels of 2000 (say less than 15 months from now).

That said...

We're totally in sync on exploiting platinum. The number 1 priority for the next release will be going head to head with Notes. We will make outlook something that we can demonstrate with a straight face to a customer considering Notes. This is a HUGE amount of work--we are 10 years behind. To date we have spent lots of time trying to outflank Notes and use lots of pieces. I want to propose that we go after the Designer/Domino head-to-head, feature-for-feature. This will mean we use up a lot of time/developers "catching up" but I think we have no choice. I am comfortable doing this if we can go into this knowing it will take several releases. I think we need to do something radical, yet straight forward. Taking the Designer and making sure we can do those things more simply in Outlook (offering a way to hide the design tools for non-power users) is something I understand sort of.

To the degree that Platinum runs on servers at MSN or on ISPs this work gets totally leveraged of course.

I will try to spend time on this on the 10th, but our thinking on the server side of things is that it will be very hard to be optimal-- where optimal is defined as work only on Platinum. There are many issues to deal with and we want to please the most customers possible. If you think of the way server stuff is done in reality today (pending the increase in platinum use) there are very different customers, scenarios, and implementations:

LORG Groupware -- this is the Notes space. These servers are run by IT people. There are developers and IT managers. The tools are full client/server and do very rich things. This is the target for Outlook+Platinum, with the rest of Office and VB as points of strength. The main scenarios are mail, calendaring, and "notes applications". We will start going head-to-head this release and measure ourselves by how well we can be a dev environment and have end-users do template based work--this will be an enormous effort and take several releases.

ISPs -- this is the FrontPage/Unix space. The servers are run for a significant monthly fee. There are datacenter people but the opportunities for development are much smaller -- in some sense the ISP is an ISV who assembles offerings (i.e. they want FP to support commerce). This is a huge asset for us since it allows people wishing to create a web presence to use office. If the web-services (next) get rich enough they could squeeze out office, but given the flexibility and richness of managing a FrontPage web this is a longer term risk for anything more than a few pages. Today ISPs prefer the Unix extensions since they prefer to run Unix-- we have both and continue to since I think absent the ability to change the server OS we'd rather have people be able to save their documents.

MSN/HotMail/Services -- this is the service space. These offer much more directed services with little or no customization. These are things that you can decide to use in 2 minutes and stop using in 5 if you choose. They are getting much richer and we want to be in this space. I don't think it is realistic to have the frontpage code run on MSN for two reasons -- first it is a different feature set than these provide (it is for web authoring, not quick communication, community, or support for roaming/sync data) and second, MSN is not an ISP but just a web service. I realize this is a fine distinction but it is really about how you approach the relationship and the software I can use. MSN will have to provide almost exclusively an out-of-the-box experience since that is the only way to have mass appeal and be scalable and reliable. We are going to work with MSN and come up with a set of features that will be Office-specific when we release.

Now the challenge we face is that it seems like we're doing the same thing 3 times. I think we can manage that, but I want to be realistic about the customer scenarios and marketplace. We can make the bet exclusively on Platinum, but I think we'll miss out on the chance to cement leadership (with FrontPage) or lead in a new area of services. It seems unlikely that we will be able to get broad ISP support for Platinum in the next 24 months--broad enough to counter the increase in users going to all of the alternative

community sites. My ISP, for example, has NT but charges twice as much and probably has 200:1 unix:nt in their customer base (they have 100,000 customers that get 30MM hits a day, hosted on probably 15 Sun boxes--not sure how much NT, but it could be equal).

We would spend a great deal of effort to have a common feature set across these -- for example the absolute minimum is that file open/save is the same (with Platinum obviously being a superset) and we would have the same look and feel for all our created HTML pages. If we did the work, we would syncronize handheld data the same on MSN as on Platinum. Those are examples. There are some specific value added things in each area:

- Platinum/Outlook -- this is the full development/customization environment. It relies a lot on IE5. It makes it easy to create a tracking list or an end user could create a form and folder from scratch as easy as they can in Notes. Workflow is in here because of the platinum store. This is a monsterous amount of work.
- FrontPage -- fastest way for a person to become a .com site. This is essentially the full roll your own case. ISPs are the primary target though for non-exchange customers this is our primary entry into the corporation. Mail is provided by the ISP and is used only at the basic SMTP/POP3 level. Customers are creating lots of pages/documents and are publishing. THere is collaboration, but in a way more like "I write this for all of you to read".
- MSN -- this is the community, roaming, lightweight service for anyone/anywhere. You can save office documents (share them?), sync your schedule (share it?), and do msn community things. You can do something like create your own distribution list or phone book.

This trichotomy absolutely does not preclude Platinum and NT from totally dominating -- and if they do it is a total win for us. The pragmatic view is to invest in FrontPage because it is a tremendous ISP asset and in MSN because I don't think Platinum will be there in the time frame you want. I personally think all three will coexist for a long time because of the differences in customers (the IT manager, the ISP, the end-user).

Well, I'll give explaining this a shot in a couple of weeks.

In terms of data -- we already do a lot of what adam says but without XML. It really isn't a big leap to assume that if a component produces XML (say the server) it can also produce a plain HTML table view. That is easy for us to consume and very universal. I'm sure there are scenarios where the XML provides additional functionality, but I am leary of changing our file format unless we can clearly articulate the benefit. We're totally open, but worried about the format issue and the churn. Adam and I might just differ on this and we'll just work through this to do what makes sense.

Nevertheless, we will definitely be focusing Access on SQL. We made some progress but we have lots left to do. A key part of this mission will be to understand and make real the schema work--but only to the degree that we can do this without churning the OLEDB infrastructure. If we have to take on yet another data access story we should rethink things--we can't afford to do this again for more ways than I can count. We definitely want some significant ERP progress from Excel and Access and if we can use the schema stuff to get us there that is excellent.

You can think of Access/Excel as the partners with ERP and the competitors to Oracle. Outlook competes with Notes. Word, PowerPoint, FrontPage are the unique assets for creating information, which no one else does (ok we have three -- a little humor this morning).

NetDocs is a big topic and I've used up by quota for the day.

There are three topics we might want to brainstorm sometime:

a) Office exploiting Platinum to lead in Collab including Platinum as an Internet service. This requires a strategy for annotation/community some of which Darryl has written about.

b) Office and our plans to be the leader of a parade in how people handle data/metadata. This is still quite vague but is the thing I want to make real to have something from Microsoft more relevant than JAVA. Is there somethign simple we could do in the Platinum release? Does it relate to the "Notes development competitor" environment?

c) A discussion about Netdocs

I am not sure what we should do in any of these areas but I would love to brainstorm about any of them.

You mentioned in the last review your desire to _leverage_ Office as a way to view/manage all this information flowing around the web. I heartily concur. I'd like to just make two points:

1) If Office _itself_ were a valid XML format (meaning that our XML parsing component could read in any Office document and create any XML document) then it would be easy for middleware applications to dynamically _pull_ key data out of office to move around the web and to dynamically construct office documents from that key data. Remember, our components are designed to run fast and lean on the server. For example, given some medical data, I could directly build the spreadsheet to view it. I wouldn't have to have Plato and pour data into it. I could just build the right spreadsheet. This is a much easier and lower-tech way for our customers to use Office to both view XML and to act as an XML source than one that requires our custom stores to talk to Office through OLE DB. It means even Unix Servers can and would build Office documents and treat them as sources of data. We're 95% of the way there in office today. I strongly advocate a public commitment to get to the last 5% in the next release. I think you'd be surprised how much good-will this would get us and how much it would cement Windows as the client.

2) Office does need to start thinking about consuming XML as a data source. XMl innately provides object state and semi-structured state, not just tables. Our tools right now (Access and Excel) are totally focused on consuming tables. In my opinion, this is a two-tier model and will be increasingly less relevant over time. I want to be able to use Excel to point via a URL at any XML provider anywhere and load in the information. We could easily push annotations to the schema of documents to describe how to do this (we deliberately made the schema extensible) and/or build XSL style sheets to pour the XML into Excel. Similarly, I want Access to dynamically build the right UI for viewing the XML data through Forms so that I can scroll through interesting parts of the data, find things, sort by elements, and so on. Ideally, Access would leverage our in-memory data-set of XML to the hilt.

Please consider these ideas as the next version of Office is planned.

Thanks
Adam Bosworth