

Amy Johnstone

From: Paul Maritz
To: Bill Gates
Subject: strategy
Date: Wednesday, October 13, 1993 4:59PM

I am sorry if I have not done good at communicating status to you (althought the thinking of using of providing the GLE2/Component Forms based Shell environment on Chicago by doing a second release of Chicago with OFS, was communicated via the email, attached at end of this message, which went to "all concerned" - see "to" list), but the issues and tradeoffs ARE complicated. I would appreciate being able to go thru this face to face - is there possibility of meeting Thurs/Fri/Sat?

I have been doing following work to get a handle on things:

- getting bradsi and jimall in sync (more below),
 - met 1-1 with PeteH (week before last),
 - communicating with other divisions (had meeting with key folks from DAD, WGA, DDT last Friday pm),
 - met on Monday with REN folk (following your mail).
- I have been trying to achieve following:

1. Recognizing that Chicago is THE next ISV target, then:

(i) Ensure that the API's exposed by Chicago are as close as we can make to the OLE direction we want to go - mainly can we get shell extension API's to be OLE (answer: yes, we can by using lighter weight OLE implementation for just those scenario's that Shell uses - ie. not for general in place editing, etc.). This will mean that any Chicago UI exploitive apps would work decently on Cairo - ie. no need for ISVs to do different work to run on Cairo.

(ii) Contain Cairo to meet following criteria:

- be proper superset of Chicago (all apps, including Shell extensions, must run)
- be available within six months of Chicago - this means that Cairo team has to have high quality beta that we can broadly give out when Chicago ships.

To do this, we have to figure out how much can/should be done in Win NT 3.11 - should we try to do more to sync up with Chicago API's, or do less and try to ensure prompt shipment of Cairo (defined to meet criteria above). I have discussed this with Jimall and he is currently working through the issues with his team, and deciding what is on critical path and can/has to be cut. This does raise the issue of whether we should simply declare Cairo to be "sync up with Chicago" release, or whether we should imposed constraints above and let Jimall decide how much he can afford to ship. I am currently working through the latter path.

2. Understand role that REN plays. I was actually impressed by the meeting with Brainmac & Koss, I think it is great move to have them be part of office. However, it was somewhat frustrating to see the overlap in what they are doing and what the Cairo shell team is doing - both are producing a Capone replacement, both are producing a set of standard views for looking at heterogeneous objects, etc. In a perfect world, the following would be true:

- REN would be build using Component forms (can be done if REN will accept Q1'95 ship date - we took action to try to convince them that it would be OK to take this dependency)
 - REN currently wraps the current non-OLE Chicago extensions, with their own OLE based extensibility mechanism - this should be the same as that officially used in Chicagi/Cairo (action taken to see if this can happen),
 - REN would use DAO wrapper for LMS, meaning it would work without change over OFS (need to get DAO closed for this).
 - REN would be "vend" some views to Cairo - why both do the job of providing standard views? This provoked fairly strong reaction from Brianmac that his charter is to "help Office, not Systems" - but I still think we should look harder at saving effort.
- I regret not having met before with the REN guys. This needs more thinking and understanding.

3. Meet with PeteH and co. to understand what Apps / Systems Plan interaction should be. I gave them almost identical feedback to the mail you sent out, namely that I think it is crucial that they get "reasonable" Chicago based releases out within 90 days of Chicago, with:

- 32bit
- reasonable Chicago UI exploitation (I told them that we guarantee that those apps run on NT)
- some way to claim that they exploit multi-threading

The hard issues then comes in when there next MAJOR release should be - should they do it in mid'95, or

Page 1



MS7048981
CONFIDENTIAL

MS7048981

should they wait for the full Component Forms UI & OFS to be Chicago in late '95/early '96?
This issue is complicated - I don't think I can have MAJOR ISV CONTENT releases much closer together than every 18 months, i.e. if Chicago is major ISV content release (Win32, UI, m'threading, MAPI, etc.) in Q3 '94 - then the next such event can't be much before late '95/early '96.

So, as best I can wrap my mind around things now, our systems/apps strategy could be:

Q3 '95 - Chicago: MAJOR EVENT for Systems, we do as much as we can to make it a major event for Apps

Q1 '95 - Cairo: sync NT up with Chicago

Q2 '95 - REN & Component forms based Office release? Major feature of this would be REN.

If we can sync

up the issues listed above, this could also work well on Cairo as well as Chicago - it

could even be billed as first OFS exploitive app when on Cairo, and be billed as

providing OFS like function on Chicago (Apps leads way)

Q4 '96/Q1 '96: MAJOR EVENT for BOTH SYSTEMS And APPS - OFS exploitive on volume platform, etc.

4. Get rational plan for EMS on Cairo. Given above constraints it means basically that we should not try to replace EMS server in first release of Chicago, but make sure that the Cairo UI environment which replaces Capone, can work well with a EMS backend (a necessity anyway since EMS store customers will not upgrade quickly) and handle EMS special message types correctly. EMS would be fully sync'd up with Cairo (OFS, DS, etc.) in late '95/early '96 (9 months after Cairo initial release). This raises issue of EMS not initially using OFS based directory.

Here is the memo on systems release priorities that I am working on. I have sent it to Brad and Jimall so far to get their reactions.

<<File Attachment: OBJS1093.DOC>>

From: Paul Maritz

To: Bill Gates; Mike Maples; Pete Higgins; Brad Silverberg; Jim Allchin; Darryl Rubin; Roger Heinen; Tom Evslin; Bob Muglia; Christopher Graham; Chris Peters

Date: Wednesday, September 29, 1993 3:21 PM

From conversations with Billg and others, it is clear we are somewhat out of sync. on the interaction of systems plan and desktop apps plan - we need a meeting to update each which I will try to get set up for next week. But here is quick status on some things from "systems" end:

1. OLE-based Forms:

Jimall is prep'ing some mail to send on this, but the OLE-based Forms package can be made independent of release of Cairo, since all it really requires is OLE. Thus usage of this forms package is something we should be considering for products that ship in early '95.

2. OLE-2 Based Shell and environment on top of Chicago - codename "London":

The major issue here had been the way in which OFS storage services which the Cairo environment presumes, would be provided on Chicago. After analyzing the alternatives (doing nothing, improving FAT, using some form of LMS, porting OFS), it was determined that the best alternative would be to port OFS - since it offers maximum benefit, and could be done with a reasonable effort (<5 man-years to provide a complete environment - i.e. moving CairOLE to Chicago, moving DFS to Chicago, enhancing Chicago IFS to allow simple port of OFS), and working set would be reasonable (worst case would increase working set by <3 MB for CairOLE + DFS + OFS + Cairo Shell) and would allow a reasonable 8MB solution.

The issue with the above would be that it would require a "new" release of Chicago, which would mean that we would probably do other cleanup, enhancements at the same time. If we wait until Chicago ships before starting on this, it means a delay of 18 months or so after Chicago v1 ships. This means that we have to start the effort before Chicago ships. Fortunately it is fairly well confined/defined work. The Cairo shell pieces can and are being engineered to be well-behaved "Win32/OLE" applications which means that it should be a port. We will look to try start the Chicago scaffolding work (IFS hooks, etc) soon.

However, presuming we can come up with a reasonable staffing plan, this would mean that the systems plan would look something like:

(adding a quarter to everything for pessimism's sake - all internal dates are a quarter

earlier)

Q2'94 Win NT 1.0a (includes sufficient of Chicago Win32 API enhancements to allow ISV to target both Chicago and Win NT in one effort)

Q3'94 Chicago

Q2'95 Cairo

Q3'95 London

Note: the above plan is not yet "committed" - we have to figure out staffing implications, so please do not forward this mail to all and sundry.

3. DAO

Dave Fulton has chartered a task force to drive the DAO interface to the point where it could be sufficiently defined as our general container access API, and Rogerh has declared it high priority. Cairo people are participating, and I have asked them to get darrylr in review loop. We should get update on this next week.

4. EMS

There is proposal on a "middle cost" road that we will have to take on providing the coexistence/migration facilities for EMS and "CMS" (the "Cairo Mail System"). We should have the costs by next week.

Systems Release Strategy - Draft: 10/10/93

This memo deals with the current issue, and the options, we face in our priorities for systems releases.

Note: Others have seen these issues earlier and more clearly than I have - jimall, bradsi, and jonl all have the right to say "I told you so".

Background "Facts":

1. The current center of gravity in the installed base market is 4MB, and the center of gravity in the new machine market is beginning to approach 8MB. It will not be until 1995 before 8MB is center of gravity of installed base, and 16MB the center of gravity of new machines (and laptops will lag this trend, just as they currently do).
2. Thus in the market place at the current moment, we face the following two major problems: (i) Windows NT does not run in 8MB, and (ii) Windows 3.1 (which does run in 8MB) does not multi-task well. These two facts are what are currently undermining our ability to say that Windows family covers most of our customers computing needs, and is what is leaving OS/2 a major opportunity to become entrenched (as an "8MB desktop OS that does multitask"). We should not panic, but we should be very focused on this as a very real threat.
3. As a consequence of not running well in 8MB, NT sales will be anemic in CY'93 and CY'94 - we will not likely sell more than 250,000 units worldwide in FY'94 (counting the 80K PDK sales). While NT 3.11 (forecasted for May'CY94) will improve this situation (NT 3.11 will most likely get NT to run acceptably in 12MB, and well in 16MB), it will not alter it materially.
4. Cairo will not lower NT's resources requirements, and may in fact increase them.
5. As a consequence of being outside of the "mainstream" in terms of machine resource requirements in CY'93, CY'94, and probably even CY'95, and thus being low volume in these years, it will be impossible to get broad ISV attention to function that is unique to the NT platform.
6. For this reason, we will have to realize that we have only two major "ISV events":
 - (i) Chicago in Q3'94 (when machine requirement for high volume is: runs in 4MB, runs really well in 8MB)
 - (ii) Successor to Chicago in late '95/early'96 (when machine requirement for high volume is: runs in 8MB, runs really well in 16MB).This means that NT/Cairo will have to derive its broad ISV support from the above two platforms, and consequently the priority of running Chicago and "successor to Chicago" apps must be very high for the Chicago and NT/Cairo groups. Beyond the server, there will be few other NT/Cairo unique apps.

More background: Objectives:

As a division we have the following objectives (in priority):

1. Above all, maintain desktop market share.
2. Increase revenue and profit by raising the per PC revenue that we get from \$35 (on average) to over \$50 (on average) over the next 3 years.
3. Keep the ISVs busy implementing new Windows functions, and the cloners on a treadmill.
4. Grow our market share on the server to over 30% over next 3 years.
5. Establish new functionality to allow us to accomplish above goals, and prevent "middleware" from reducing our OS's to "graphical C> prompts":
 - support for workgroup computing
 - distributed systems capability
 - object oriented programming paradigm
6. Make things simpler for the end-user:
 - make the PC an appliance

Implications of the above objectives:

1. **Market Share:**

Since our most immediate issue is a possible loss of market share to OS/2 on the desktop, we should focus on the most immediate way of addressing it - i.e. get Chicago shipping as soon as possible. Brads and I must be willing to lie down on the tracks over this one. Beyond its currently committed set of functionality, and subject to any major performance/size/useability issues, there is nothing worth delaying Chicago over. This will become a major issue internally as we seek to make Chicago our next "silver bullet for all ills". It means we need to be prepared to decide how to deliver things like Russ's Online client & Paulo's Multimedia stuff in an update release, or through alternate means.

What does it mean for Chicago to stop OS/2 dead? I believe we have to do following:

- (i) Provide a smooth multitasking environment - this does have customer value, and IS what is currently selling OS/2. To do this we need to get the focus moved to 32bit applications. This means that it is critical that the MS applications group (at a minimum) have a 32bit version of Office ready to go with 60 days of Chicago.
- (ii) Perform very well on 8MB (better than OS/2), and OK for existing scenario's on 4MB.

- (iii) Have a "cool" image. I believe that our new Shell, Plug n'Play, and Remote operations capabilities will do this.

2. Increase Revenue and Profit:

The most immediate opportunity to do this will come from increasing net per PC revenue on Chicago. The basic strategy here will be to:

- (i) Offer OEMs following options:
 - a base ("silver") version of Chicago at same price as DOS/Windows today (approx. \$35)
 - offer them a Premium ("gold") version at a significantly higher royalty rate.
 - offer them pieces of the premium versions for extra royalties (over the base rate), so that OEMs can selectively enable hardware features.
- (ii) Offer end users an upgrade package that upgrades any version of Windows (past or present) to Premium ("gold") level. The net retail revenue of this package would be equal to or greater than the royalty we would receive from an OEM.

Figuring out this packaging such that it will be acceptable customers and OEMs in terms of value, will not cause us to lose market share, will not be used against us by our competitors, and will incent ISVs, is a large and key challenge for us.

What strategy to pursue for NT? First yet more background:

The following are all priorities for NT/Cairo today, we need to decide which ones to optimize for, in what time frame. I list them out first, before giving some recommendations:

1. Compete for market share vs. Novell:
Since this is a priority for us, and since this is something that clearly Windows 3.1 and even Chicago cannot address, this should clearly be a high priority for the NT effort. We should probably say that until 16MB becomes "normal" (late CY'95/96), should we say that this should be the #1 priority for the NT group? If so what about Notes (see next objective)?

However, competing with Novell is complicated in that, as Jimalf and I have previously articulated, the way to win market share on the server, is to win it by changing the relationship between the client and server - and have the client increase the functionality and integration it expects from the server - only then can we will start to really turn the tide. This means that until we can get OFS function onto a high volume client (late '95/96), we are going to have slow progress vs. Novell.

2. Compete for market share vs. Notes.

Again this is inextricably tied up with the client. I do believe that "OFS" strategy of:

- (i) "consolidating the all the incompatible, hard to administer, workgroup databases into the file system", and
- (ii) "enabling the Shell to be the basic browsing/query/forms composition tool for the workgroup database") is a very powerful one, but it will not work until we have this function (OFS and the OLE based Shell environment) on the high volume client.

3. Defend against OS/2:
The goal here was to use NT's "real OS" capabilities vs. OS/2. Owing to memory requirements as noted above - we will be able to use NT only in limited situations.
4. Defend against UNIX on the high-end desktop and the server:
This remains a viable goal for the server but it will mean having to do a lot of the server apps ourselves (at least initially) to overcome momentum and hostility from existing UNIX server ISV (see NT as commoditizing their market). On the client we probably have to pay more attention to "workstation" technology (e.g. 3D) - but even if we are successful, the numbers will be small.
5. Pioneer and introduce new software technology:
e.g. Win32, DFS, DS, OFS, CairOLE, Components Forms, etc. The issue here as noted above, this will not matter to ISVs, or affect the Novell/Notes war until they are exploited on a volume client.
6. Support new types of hardware:
Windows NT was engineered to support new hardware: SMP and RISC. SMP is important today on the server, and may become important on the client in '95/'96 (Intel is positioning P54C - 0.6 micron Pentium - to be used in pairs, to provide an upgrade path for customers).

RISC could become important, but not until the next generation of RISC. I.e. to be brutally honest, in the current contest between R4400, HPPA, PPC 601 and Pentium - it is safe to say that 486/Pentium/Intel clones will be the winners on the PC desktop. To the extent that RISC is important in near term, it will be on the server. The next opportunity for RISC at the desktop will be in the timeframe of the P6 vs. MIPS R10K (a.k.a. T5)/IBM PPC 620 battle (CY'96). To win, they have to open up a > 2x advantage.

The above points paint a fairly bleak picture of our investment in NT hosted technology being two/three years ahead of its time in terms of giving us a decisive market advantage. We face triple whammy of (i) not deriving decisive competitive advantage from it during this period, (ii) having to fund it (550 people in systems), (iii) having to explain and market it.

So what should we do with respect to NT/Cairo?

First, I think we should consider what our medium/long term (CY'96) endpoint should be. I believe that this should be a Windows client and server that do the following:

- support the functions needed for us to compete long terms (new file system, and UI and programming environment that exploits it),
- is truly scalable across hardware (i.e. we have one client SKU that covers range of interesting hardware).

In this timeframe, the concepts of NT, Chicago, Cairo disappear for all customers and ISVs (exception perhaps of device driver writers). We are back to "standard" and "enhanced" mode. Customers get one SKU for client, one SKU for server, and maybe decide to enable some extra functions by paying more.

When can we realistically achieve this? If we say that the scalable client is based exclusively on NT ("plan of record"), then realistically this going to be when a Pentium class machine with 16-24 MB of memory will be both high percentage (> 60%) of new machines, and greater than 30% of installed base - this will not be until late CY'96/97 (i.e. 3+ years from now). If we want it before then, it means that we have to have an "8MB client" (with OFS, and associated UI/programming environment) - and I believe this means that we cannot rely solely on the NT base to deliver this, and must host the next set of key client functions on Chicago as well (latest code-named for this release is "Memphis", but has also been called "ChiCairo" and "London").

Key features of NT/Cairo:

1. NT base functions (portable, secure, high-performance server, SMP):
We probably do not need these for the client until CY'96 - and then only high-end clients. It is OK to leave these "16MB" unique.
These functions are crucial for the server, and we should invest to keep them competitive vs. Netware, UNIX, Workplace OS.

2. API's:

Given discussion above, we will have the following two generations of API's:

- (i) Q3'CY94: "Chicago":
The Chicago generation of Win32, and the EMS generation of "MAPI" & "Capone message types" (I don't totally understand the latter). We have to make sure that there is a release of NT (call it "Cairo") within 6 months of Chicago that definitely runs the Chicago generation of Win32/MAPI/Capone message apps - all of them. Given this six month constraint, we need to decide how many other objectives we can accommodate:
 - how much supersetting of rest of Chicago can or should be done (e.g. Plug and Play),
 - what can be done to improve combination of EMS/NT combination
 - how much next generation (see next point) technology can we introduce on NT (DFS, OFS, DS), and how much of this technology could be exposed to the Chicago in this timeframe (e.g. could we at least release DFS for Chicago at same time as Cairo),

- how much can we allow into NT 3.11 without jeopardizing above.

- (ii) Q1'CY96: "OO Environment on Chicago" (or "Memphis")
This is when we take functionality currently in Cairo, and put it on the 8MB client - i.e. OFS, Cairo OLE, Component Forms based UI/Shell, Component Forms based end-user development environment ("CDE") which competes with Notes's end-user environment. This combination of function replaces to a large extent the MAPI/Capone messages environment.

We can make two decisions about this "OO Environment" function:

- (i) Subject to getting Cairo out within six months of Chicago, Jimall can decide how much of this function to put into Cairo as get it done early, allowing us to have it tested for volume deployment in CY'96. But until then we should be under no illusions as to how this would help vs. Novell/Notes.
- (ii) It is confusing to even have to speak about it until we have it on the high volume client - until then (Q1'CY96) we will just have to take our lumps and use NT's base function to compete with Novell, and EMS/MAPI/Capone to compete with Notes.

Options for public positioning NT/Cairo before Q1'Cy'96:

Given above options for NT/Cairo priorities, our public position can be:

Option A:

- MS's UNIX for the Server (scalable, secure, SMP, etc.)
- A secure version of Chicago for the client (which requires 16MB).
- Technology introduction platform for new API's and distributed systems features.

Option B:

- MS's UNIX for the Server (scalable, secure, SMP, etc.)
- A secure version of Chicago for the client (which requires 16MB).

Recommendations:

1. Ship Chicago ASAP to hold our market share - don't hold it back for other objectives.
2. MS must ship 32bit Chicago apps within 60 days of Chicago to support Chicago. We need the applications group to re-align around this, and consider implications on this like component forms.
3. Adopt decision (I) and positioning A on NT - but do not let priorities get confused. It means we must close all remaining issues on API that are exposed in Chicago, and which have to be supported in Cairo.

4. Ship Cairo within 6 months of Chicago (i.e. no later than Q1'CY95) with 100% superset of API ad UI content- ensure that we have 95% confidence plan to do that.
5. Articulate clearly where we are going: scalable client and server in Q1'CY'96.

SECTIONS NOT COMPLETE

Interaction with MS Applications Release Plan

Current MS apps plan is:

- do minimal update releases for Chicago with 60 days of Chicago.
- do a major release in mid'CY95 based on Chicago API set and Component Forms.

Interaction withn MS Tools Release Plan:

The Cross-platform Challenge

Both Novell ad Lotus are mounting an explicit cross-platform challenge to us. How should deal with this?

Organization/Morale:

Are we organized optimally to do all of the above? What are morale implications for Systems (esp. NT team)?