

To: Mike Maples

From: Tom Evslin

Date: August 25, 1993

Subject: Ideas for WWPG Organization

What are the Products?

Applications and Systems?

Our traditional way of packaging functionality as either applications or "systems" seems to be breaking down. Prices for individual applications are eroding. Cross application functionality is becoming an important selling point as in Smart Suite. Some applications like mail require companion serverside processes. Some products need to be sold to ITG or network management rather than the individual user.

Cairo in particular and object oriented technology in general require that application functionality be available in applet form to allow a document centric view of the workspace.

Needed cross application functionality like forms and a common data store don't get done, partly because they get enmeshed in arguments between our application business unit and partly because they are too big for one business unit to pick them up and run with them.

The boundary between system and application is becoming less and less clear. So we may need a new model.

MegaProducts?

One suggestion has been that we drastically reduce the number of MS SKUs (except in consumer division). Our history with this has been mixed. Office is a huge success but Office Pro isn't selling at all. Windows for Workgroups has been a big disappointment despite the fact that its components Mail, Schedule Plus, and Windows are selling very well. Works is significant in the entry market. The components of Office still have individual identities and perceived value as separate applications. We don't know how well Office would sell if it weren't perceived as a cost effective way to obtain products with an established value and established prices.

MegaProducts make it hard to justify world class feature sets in any one product. The ad for a MegaProduct can tout integration and some very high level functionality of components but that's about it.

MegaProducts will be almost impossible to ship. It is nice to think of all the components converging on a ship date but, in practice, there will always be a critical path for one component which delays the release of needed features for another component. Moreover, integration and regression of too many components will add months to the release cycle.

The MegaProduct strategy is a dangerous one for MS even if it succeeds - perhaps especially if it succeeds. If we had one product which included all desktop apps and a full client shell and managed to establish that as the way people buy software functionality, we would have no competitors since no other company in the industry is capable of putting together such a product. We then would face the twin dangers of very intense government pressure and internal laziness and unwillingness to innovate due to dominance. I don't think the strategy would succeed so I'm not much worried about this; but my point is that success might be as bad as failure.

MS 5048598
CONFIDENTIAL

Applets?

Certainly once Cairo ships, much of today's application capability will be delivered to the market as applets which are active when appropriate in the document creation and editing process. It is important that MS think through what is involved in selling and distributing applets so that we do our part to assure that there is a healthy supply of these from third parties.

We will also have to make sure that the functionality of our own apps is available in applet form as we are doing with OLE servers.

But it is hard to imagine Microsoft's application product line consisting of thousands of applets. Each product would be too small for either our sales force or management to concentrate on. Small companies will deliver small apps, quite possibly through Microsoft Online or some other communication based service or by unlocking items on CDs containing applets from many vendors (which we may choose to distribute); but I don't think MS will bring product to market as individual applets.

Suites?

Our one clear success in repackaging is Office as a suite of applications with established identities and value. It is pretty clear that sales of Office help establish Microsoft desktops in both the Windows and the Mac market. Word and Excel lend each other strength in this arrangement as corporate standards. In this sense, sales of office may also spur sales of individual copies. It is not as clear what value Mail and PowerPoint add. It may be that the mail licenses distributed with Office help sales of mail servers if we handle this right; but we have yet to establish this.

It is fairly clear that the Office must add better integration between its apps in order to maintain both price point and market share.

There is no evidence that simply adding another product to this suite either increases penetration or perceived value - e.g., Access.

I think, based on limited data, that our products of the future will be packaged like Office but will continue to be available as individual SKUs. The applications themselves may consist of packages of OLE servers and other little pieces; but I don't think we'll sell them that way. When installed, there may be no single new icon on a desktop; but there will be a clearly defined and complete set of new capabilities after installation - e.g., word processing, number crunching, mail, scheduling etc.

Conclusion

I think we will sell primarily suites but that the individual "applications" which make up suites will also be available. Taking the example of Office, we will organize a major campaign around this. However, as part of that campaign, ads may run which feature the word processing or number crunching apps individually, not because we want to sell them that way but because that helps establish the value of the package.

Other suites might be Messaging including mail, scheduling, rules, document library and remote support; Toolkit including various compilers, build environments, and prepackaged objects and sample code; Database; and Windows including network access, device sharing, information sharing etc. The individual applications like Mail and Schedule Plus would continue to be available separately and might rev on separate schedules sometimes but campaigns wouldn't be organized around them. Moreover, once purchased, components like the mail client might disappear completely into the shell and just add functionality. Some suites might have turbocharging add-on packs available.

Each of these suites really has an "application" and a "system" component. Office in the future has its shell as the system component; Windows has the client - Explorer and its descendants - and its file servers, print servers, indexing etc. Database has Navigator, Access and SQL. Tools have the server which does versioning, backup etc.

MS 5048599
CONFIDENTIAL

My guess is that we'll end up selling the server functionality a desktop at a time rather than one price for the server and then separate client pricing. However, we may do the actual metering of use at the server to avoid losing revenue to other clients written to piggyback on our server.

Organization

Assuming that the suites above are the proper way for WWPG to deliver its product, the next question is how we ought to be organized to build and market them. It is certainly possible to think of an organizational structure for marketing completely separate from the development organization. Moreover, developers could be organized by the type of code they are writing - client, server, network etc. But I think this loses much of what had made Microsoft successful so far: a focus on product rather than just on technology and a close association between product marketing and product development.

On the other hand, we could be organized into very many business units, each with a single product and bottom line responsibility. New business units would be formed constantly around new ideas and unsuccessful business units would have to be ruthlessly trimmed. This is something like the way magazine publishers work - always starting new magazines, killing most of them, and running with the successes. The trouble with this organization is that it makes it hard to obtain synergies between products, almost impossible to manage putting together suites of products many of which have to rev at the same time, and there would be so many business units that they would have to be collected into groupings of some kind before reporting to top management. Following this structure, it is also very hard to ever get technologies like forms done which benefit many products but impose a cost on a single business unit. I think this structure could work and is a natural outgrowth of what we've done before; but we can do better.

I think we should reorganize in a structure that looks like our current divisional structure with some key differences.

- Each division is responsible for developing or acquiring the components of one or more suites and is measured by the marketplace success of those suites and their components where sold separately.
 - All outbound marketing is at the divisional or the WWPG level.
 - Divisions are organized into product rather than business units. In general, these units produce applications - Word, Mail client etc. - but they may only produce technology components like forms, content search engines, drivers etc. or do a particular job like Ireland.
 - When a technology is needed across divisions, it will normally be assigned to a product unit in a particular division for implementation. However, the cost of that product unit gets allocated to all the divisions as overhead while the technology is being developed.
- or
- There could be product units reporting individually at the WWPG level for major pieces of technology.
 - The divisions get measured at their bottom lines.
 - Divisions can set up new product units for new products or new technologies.
 - Divisions can include components from other divisions in their suites but must share either revenue or development expense.

Not a radical change from what we have now except that each division is organized around a suite which includes system and application components and that marketing is done at the divisional and higher rather than product unit level. I don't think it works to separate servers and apps with the same functions (e.g., SQL and Access, Capone and EMS) into different divisions because we always need to have a client

MS 5048600
CONFIDENTIAL

which exposes server functions and servers which support good client functionality. Especially if we get our revenue by selling client access, a single division has to make the tradeoff decisions of what to spend on the server in order to provide appropriate client capability. Office has to be able to decide what's in their shell and what's in their apps and how they interact.

Workgroup strategy, at least prior to Cairo, ought to be the responsibility of a WWPG level marketing group. This group might repackage products from other divisions to produce an MS Workgroup Developer CD or even a Client pack with some forms capability, a mail client etc. Since our strategy depends on many apps being workgroup tools and cuts across all divisions, it needs to be executed at the group level to be fully effective.

Here's the products I'd put in the divisions - this does not imply anything about how the divisions are organized internally.

Windows

DOS

Windows

NT

Cairo

Platform support (e.g., WinPad, non Intel, @work etc, Mac)

Network support

Device support

Directory (file system, network, mail, security, routing, many apps all need to work off the same central directory which should be a powerful database. could argue that this should be in Database)

Tools

Languages including VBA

Forms

Controls

Development systems

Cross platform toolkit (basically everything to need to assure that we don't have to either rewrite or drastically degrade apps to get them to run on the Mac or UNIX where appropriate)

Database

Access

Fox

SQL

Navigator

DAO

OFS?

Office

Word

Excel

MS 5048601
CONFIDENTIAL

PowerPoint

Project

Ren

Messaging (Directory Based Applications?)

Mail clients including the Cairo, Utopia, and WinPad clients

Mail servers including CMS

Scheduling

Procedural workflow

Document library (because it needs to be accessible from the mail client and leverages the directory in many of the same ways that messaging does. This one is debatable)

Fax server (incorporating EFAX technology from Windows division)

Fax client (ditto)

Consumer

no change