To: Andrewk, Duanec, Grantg, Kurtd, Ralfha, Richardm
Cc: AndySchu, DavidT, AndyH, Jonde, Bobmu
Date. February 18, 1999
Re: Excitement for Office10—Next Steps

Office10 Challenges

The purpose of this note is to sum up where I believe we are in the planning process for Office10. We've had a number of planning retreat and many discussions among program management, product planning and our team. There is much more we do not yet know and it is important not to get the impression that we have a great deal of planning already done—in particular any features mentioned in this memo are designed to illustrate the concept, not to offer a requirement. From the outset it should be abundantly clear that there are a number of key challenges for Office10, but beyond those challenges lay an incredibly exciting release. In fact, the foundation that was created by Office 2000 sets us up perfectly to accomplish much more in Office10—we will be able to further our investments in groupware with Outlook, leverage the investment in HTML and web server to advance the *results-centered web*, advance our position with data access due the work in SQL and components, and because of our investments in TCO we will be able to revisit some of the areas of personal productivity and with a new perspective be able to innovate in this elusive space In short, we are well positioned to have an enormous amount of fun developing Office10!

For more information please refer to the Next Release of Office planning document as well as the wealth of information on the Office10 web site.

Today we are facing a number of key challenges. First and foremost our number one priority will be enhancing our customer relationship by supporting and maintaining Office 97 and Office 2000. We can't ship Office 2000 and then simply *move on*. We will look at ways of increasing our commitment, both in our organization and in the sales and support arena, to our shipping products. We will have more people than ever working on "QFE" related issues—QFE is in quotes because we will need to stop thinking of a QFE as something that we resist and something that is a one-off. We simply will take the amazing progress we have had to date (thanks to efforts by Grant and Alex and others) and build upon that.

We also face numerous challenges in our development process. It is worth considering the Office 2000 schedule and just for a moment we should feel some guilt:

Milestone	Original Date	Actual Date	Delta
ZBR1	4/13/1998	8/21/1998	+4 months
Beta2	5/11/1998	10/15/1998	+5 months
US RTM	7/13/1998	3/15/1999	+8 months

On the one hand we should feel pretty yucky about our failure to schedule well. And we do. On the other hand, despite measurable improvements, we have actually improved our process greatly This release has not been the death march that characterizes the last year of Office 97;

our beta releases have been **rock solid** beyond the expectations of even our harshest critics—due to the incredible efforts by testing to ensure high quality public releases that went out to an order of magnitude more people than we originally planned. We improved massively in the closure of the pre-development schedule program management milestones and the closure of each development milestone. Development earned a lot of credit for being appropriately hardcore during each of the milestones, while at the same time maintaining flexibility regarding the unknowns of the major efforts we were undertaking.

Second, not much more can be said of our daily built-test-release process, other than we will make this an all-appropriate-hands effort to address shortly after the US release goes out. We made major improvements in the release team's build room and are able to turn our all the configurations in a much more efficient manner than ever before. This area, perhaps more than any other, matters because it directly impacts the daily routine of everyone and there is every reason to make this a much less painful experience.

A third challenge I would highlight is the challenge of building the right product for the right set of customers. Office 97 surprised many people, including many at Microsoft, with its level of success. We should take a moment and pause to consider that we probably beat ourselves up too much for being behind the Internet curve and rushing to judge our product that had not yet even made it to customers' machines. On the other hand, this *panic* caused us to rethink the Office suite and to invest substantially in key scenarios of web-based documents. This investment has rejuvenated the Office product in the eyes of pundits and internally, but we still do not have customer feedback on this investment. We do not yet know if people like our web document creation tools, our web data analysis tools, and our integration with servers. We know these features demonstrate to rave reviews and they appear compelling. But we still need to learn from customers before we change things substantially. Let's consider Office 2000 a major asset that still must be perfected, but not something we must consider redoing yet again. It will take enormous discipline to avoid tweaking things in Office 2000 that bug us, but we must show this inner strength or we risk churning customers too much.

Although there are many other challenges, the last one I wish to highlight is the challenge of having synergy between Office and the rest of Microsoft. Many times we have talked about the pains of synergy and the difficulties of "working with other groups". But with Office 2000, we wrote the book on how to create mutually beneficial dependencies and deliver on them. These were hardly easy and few were without their skirmishes, but our customers can clearly see the dependencies and the benefits that come from them. We have so many examples where we took on the pain of coordinating, but lots of great features came out of this work: SQL server and Access, IE5 and user-interface consistency, Trident and consistent rendering with line services and Office HTML, TCO and Windows 2000, FrontPage and the Office Server Extensions, Rosebud and the Server Extensions and the name space extensions, etc. There is a downside to this synergy, which is that we have also burdened our customers with an incredible quantity of system software that is needed in order to upgrade to Office 2000. We followed all the new rules and have this software clearly separated and identified, but it is still a barrier to upgrading. We had the philosophy that upgrading Office would just require new system components. Although this makes creating a new feature easier, the downstream impacts of installing that feature and managing it in a corporation are significant. We will have to work harder than ever before to make upgrading to Office10 easier and less dependent on these sorts of components, while at the same time building in strategic dependencies in numerous places in the product—those dependencies are what make Office a key part of the software that runs businesses.

Core Tenets

That leaves us with only three basic tenets for planning Office10 that we should not defy:

1. **Office10 is 100% compatible with Office 2000 in every imaginable way.** This means solutions, setup, documents, files, etc. We will preserve every customer investment in Office 2000 with Office10. We demonstrated that we could do this with an incredible upgrade feature and our unchanged file formats (except for Access).

This also means that we will not tweak the user-interface unless we are spending significant energy to radically improve a particular experience, and even then only if this will advance the entire suite. We will outdo ourselves again on this dimension.

2. **Office10 performs as well as Office 2000 on the same hardware**. The work on performance in Office 2000 was impressive. To come this far in functionality with barely a nudge in the baseline and real-world benchmarks is awesome. It will be an even greater challenge for Office10 as we see customers getting new hardware and it becomes harder to justify working on "old" hardware. Nevertheless, we will need to keep the philosophy of "upgrading Office does not force a machine upgrade".
3. **Office10 takes advantage of, but does not ship or require, new *system redistributables***. We will simply freeze the level of system components we ship with Office at the set we ship with Office 2000, and we will not ship these again as an integrated part of setup. By the time we ship Office10 we suspect that most customers will have upgraded to a service pack for Windows 95/98 or Windows 2000 that supports all of the "system" components we redistribute with Office 2000. We will remove those and no longer ship them in Office10. We will absolutely take advantage of new services, but Office must continue to install and function without those services there—we will disable options or interface depending on the availability of a new component. This is a drastic departure from the past for Office, but it is a natural progression from the Office 2000 product's isolation of these dependencies for IE5. To be perfectly clear, although we will not ship these updates in Office, we will do work and have features that require updated components. Customers will get those features by installing Office10 and then installing an OS service pack. A partial list of these components includes Darwin, MDAC, VSE, IE, OLEAUT32, MAPI, and any other component developed by platforms.

We will also make this release the coolest release of Office ever. With Office 2000 we get lots of applause and excitement. With Office10 we will get even more. User-groups will be begging us to come do demonstrations of Office10. Our number one customer is the influential end user for it is that customer within corporations that drives so much of the excitement about Office.

ATG Synergy and Schedule

A high priority for Office10 will be the adoption of a shared schedule with the entire ATG product line of BackOffice, Visual Studio and Office. This is a huge challenge that we are going to take on for this release.

The key element driving this strategy is that we need to deliver an integrated product offering for our LORG customers that substantially competes with IBM and Lotus Notes offering. This means that we need to have a synchronized release of Office to match the evolution of BackOffice (particularly Exchange Platinum and PKM).

We have decided to accomplish this by defining two waves of products. Currently, the first wave is going to be towards the start of 2000 (April 15, 2000 is the current target date) and will include Office 2000 Service Release 2 (or potentially 3), Exchange Platinum, Tahoe (the code name for PKM's product), Microsoft Office Developer (the "grizzly" product) and of course SQL Server 7 and the rest of BackOffice and Office Premium. This release will be the first coherent set of functionality that we have delivered to customers for solving their knowledge management and application development needs. It is just a start.

For Office, there is a lot of desire for a "short" release of Office that ships with this wave. We have talked much and concluded that it is not feasible to turn another release of Office in 12 months if everyone works on it. We have also discussed the idea of doing a parallel development effort, but with that comes two negatives we did not wish to take on. First, the small size of the development team required would not be sufficient to do substantial work on leveraging Platinum (particularly with Outlook) and second the bulk of the team working on the second release of Office would be forced to synchronize with a set of yet-to-be determined releases sometime after the April 2000

release, but not long enough after that the other products would be able to turn their releases. What we will do is the necessary work in the service release to enable this first wave. At this time we think this is minimal because there is already a significant foundation of features in Office and what we need to do is be clever about adding the desired functionality without causing a major compatibility headache. The three features we have talked about in this area include: Outlook using a MAPI provider version of the Platinum local store, integrating the PKM file open experience (we could do this via an Office COM Add-In as they are doing already), and the Tools work on writing an Access add-in to help program solutions for Platinum. None of these require, to the best of our understanding, work in the core code of Office. Our current preferred ship vehicle for this is a separate service pack (we might think of it as an "Enterprise Update") for Office 2000 that coincides with our service release after Windows 2000 ships.

The second wave of products is a release timed for availability in the start of 2001. This is the release that the bulk of the Office team will be working on and defines the first full entry of Office as a competitor to Lotus Notes on the client side combined with BackOffice on the server. This release takes advantage of all the functionality in the first wave of products as well as the new features that will be added to those products. This seems like a long way off at this time, but it will be an aggressive schedule and will be date driven with this release. To accomplish this we are considering a schedule that is about 18-20 months from the start of development to US RTM. This means we will do one less milestone in this project and we will necessarily be hardcore about having major architectural changes span milestones. We will also need to improve over Office 2000, which was originally scheduled as a 15-month release and ultimately took 22 months on schedule.

Below is an idea of how this schedule could work. **This is not** the schedule for Office 2000 and is not final. Rather this is just a reality to check to see that we can fit a realistic schedule in this time frame. It will require discipline above all. Note that not all the project milestones are listed since we could have additional beta releases and ZBR metrics as well.

Milestone	Office 2000 actual	Office 2000 allocation	Office10 suggestion	Office10 suggested allocation
Organization in place (shared feature teams)	January 1997	2 months after RTM	March 1999	Soon after RTM, as practical
Priorities/Vision	April 1997	5 months after RTM	May 1999	Two months after RTM
Schedule Starts	May 1997	1 month after vision	June 1999	1 month after vision
MM1	May-August	4 months total	June 1999-October 1999	4 ½ months total (10 weeks of coding)
MM2	August-November	4 months total	November 1999-March 2000	4 ½ months total (10 weeks of coding)
MM3	December-March	4 months total	None	N/A
Broad Beta	September 1998	6 months after code complete	July 2000	4 months after code complete
US RTM	March 1999	6 months after Beta	December 2000	5 months after Beta
Total Schedule Time	23 months vision to RTM	Original schedule was May-August on schedule or 15 months	19 months vision to RTM	18 months on schedule coding to release with ~8 weeks buffer

So this is going to be a huge challenge. We are going to stick to a schedule like this so that we can also sync up with BackOffice for the second wave of the ATG products.

Big Bets

As with Office 2000 we will place some big bets on technologies. In many ways we will continue to bet on a number of the same things as we bet on as we started Office 2000, but we also have some additions.

Exchange Platinum. We are betting huge on the combination of Exchange Platinum and Outlook to effectively get us in the game versus Notes. A majority of our efforts on Outlook, along with efforts from programmability, HTML, and user-interface, will go towards supporting an effort to put Microsoft in the groupware space. It will take several releases of work to get us there, but we will start with a huge effort in Office10. This is the most critical competitive landscape facing Microsoft in the Office category.

Departmental Webs: With our investment in HTML and Web Server in Office 2000 we began a big bet on the value of easy to deploy and manage web sites. We will continue to bet that this paradigm makes sense for many of our customers, particularly those that are not using Exchange Platinum or those hosted by ISPs. This is a bet we made as we started Office 2000 and we will continue to move more functionality in this direction.

MSN: The third tier (see below) of server investments Office is making is the bet on MSN as a provider of very cool, lightweight, services that can be used by every Office customer who can take advantage of MSN. The challenges of providing a scalable set of features for MSN are enormous and totally new for our team. Where traditionally we focused on breadth and depth of features, this effort will focus almost exclusively on providing a small set of things that are incredibly robust and scalable.

We are not retreating from any of the bets we made for Office 2000. We bet on universal viewing (HTML) and we will continue to improve the fidelity and functionality, without changing the file format. We bet on the existing product code base and not entirely on a new product. We are not turning down our investments in Office's main code base, though we are choosing not to increase it (i.e. no new headcount). We will continue to ramp up efforts on a new code base, called NetDocs.

Next Steps for Office10

As we begin to deliver a vision for Office10, four priorities have come forward enough times that we should consider them a starting point. A vision process, as will be driven by AndrewK, will surface the rallying points for Office10 in a clear and concise manner. In that spirit the following is merely a starting point. We learned from building Office 2000 that the 6 focus areas we came up with were clear enough that very few problems arose over who was responsible for which areas. We also learned that marketing quickly took our 6 areas and merged them pair wise into 3 areas. Perhaps it is worth considering this as we plan Office10?

The areas that we should think about will allow Office10 to continue of the mission of being the best execution of an integrated suite of internet-centric communication and productivity tools for creating, editing, sharing, synthesizing, and analyzing business information. That was the mission we set out on when we started Office 2000 and we are not done yet. We will make Office10 super-exciting for all of our customer segments—and super-exciting is the operative phrase.

For this note, let us consider the following four product areas:

1. Delivering on the promises of Lotus Notes using Microsoft Office and BackOffice
2. Creating killer web sites for knowledge workers and IEUs
3. Innovating the customer's Office experience by integrating exciting new technologies
4. Nailing the fundamentals by making Office better at its core

The interest in these focus areas span our traditional customer segments:

Product Area	Core Constituency	Other Constituencies
Promises of Notes	CIO	Influentials, BDM
Killer Webs	BDM	CIO, IEU
Innovating Experience	IEU	
Fundamentals	IT and EU	Everyone!

Office and Servers

Before we explore at the possibilities for the four product areas it is important to look at the increasingly important role of servers and how this will impact the feature set, dependencies, and customers of Office10. This section offers a framework for describing how servers are viewed in the marketplace today.

Today we are in a world where there are lots of servers, but the Office product does little to unify this experience. If you want to talk to a web server then we have FrontPage and the Office Server Extensions that allow saving and opening documents to appear seamless, as well as additional features such as threaded discussions and subscriptions to enhance collaboration between customers. Office customers can also take advantage of this functionality from a broad range of ISPs. For Exchange customers, we offer Microsoft Outlook that is the ultimate client for messaging, but does little in the way of richer collaborative work. Microsoft offers an ever-increasing wealth of services on MSN, which Office does little to take advantage of (though Office Update is making inroads at increasing the value and support of Office post-sales.) To our customers, there is an array of servers, from Microsoft and elsewhere, and Office has inconsistent connections to these valuable resources. Our challenge with Office10 is to provide a spectrum features that crosses through the variety of servers.

We will also continue to bet on SQL Server as the primary server database. Through the efforts of OLEDB and our development tools we will get closer to a unified view of both structured and un-structured data. We will work to make the experience of using these two servers more unified for developers, but until there is more of an API synergy we will be in the position of supporting both uniquely.

An interesting way to think about the variety of servers and services available to Office customers is to think about example scenarios for each of three "server spaces" of groupware, webs, and services. This is how they exist today and how customers are using the servers today.

- **Services**: Consider two people with a business idea. They begin by setting up a project space perhaps on eRoom or MagicalDesk. There they exchange files, maintain mail addresses, and perhaps manage some contacts. It took them almost no time to set this up, all access is through any standard browser, they can view their information across any platform from any location, etc. HotMail is Microsoft's entry into the web services space and we can expect a wide variety of these services to become available including chat, buddy lists, document storage, project spaces, etc. There are literally dozens of web sites offering some combination of *virtual office* services including eRoom, Jump, Yahoo, Excite, Visto, and MagicalDesk. These services are characterized by near-zero cost of entry and trial and little customization. If you like one of these services you spend only a few minutes learning it, and if you don't like it you had very little investment and just move on to another one.
- **Webs**: Now consider our entrepreneurs as they receive some funding for their company. They have moved beyond exchanging some information to requiring a presence on the Internet. They go to an ISP and purchase a .COM site, or perhaps they settle for members.tripod.com/mycompany and use TriPod. Using the FrontPage server extensions they are able to set up information about their company, collect names and addresses with a database component, and if their ISP provides commerce they can even accept orders. Web sites are the basic world of HTTP servers. Unlike the services

space, webs are run by departments or by whole corporations. Webs are characterized by the fact that they are much like today's file servers for sharing information such as static documents though they are accessed through web browsing technology. This is the target customer for FrontPage and Office 2000 adds a whole range of document creation along with easy saving and loading of information to these servers. Webs are also the preferred offering of today's hosting ISPs who also create the domain name and manage your organization's Internet identity.

- **Groupware**: Our example company is now thriving and has now grown to 200 people. The company has all the problems of any medium sized organization—there are specialized departments, communication needs to happen across multiple locations, forms need to be filled out for various business functions, and the organization can now afford to have someone spend time developing a custom application (perhaps a customer tracking solution). Today, this is a space characterized by Lotus Notes/Domino (or specialized software, or no software). This is a world where LORG and MORG customers install and administer their own servers. There is a vibrant developer market centered on customizing solutions for this space. Exchange Platinum is targeting this space as the preferred server. Outlook is the preferred client for Platinum but is missing many features to be comparable to the Lotus Notes client or the Domino Designer. Although Lotus is working to provide a seamless experience between using their Notes client and a browser, developers and users must be aware of their target.

One thing that should be obvious is that this segmentation is based on today's technology, but just as much, if not more on the customer experience. It is clear that Lotus intends to drive their product through this space all the way to the service world—and as this note is prepared we leaned about an arrangement between Lotus and AOL using eSuite for small business. Given the implementation details of this challenge, it seems that it will be some time before there is the richness of Lotus Notes available as easily as a web service. That is not to say that a current company offering a web service could not enhance that service to meet some specific Notes scenarios. As with any segmentation, it is not long lasting and it is important to consider carefully what investments we make based on such a short-lived separation.

Features	Services	Web	Groupware
Setup	None. You just go to a URL and begin using the service after a brief registration.	Sign up with an ISP or install a server with the FrontPage extensions (or even just FTP). Moderate learning curve as users try to understand the subtleties of the HTTP namespace.	Extensive. Involves touching client and server as well as specialized server infrastructure for mail and for administering the server.
Example Features	• Browser-based mail • Contacts list • File storage and sharing • Project space	• Corporate presence • Saving and loading HTML files • Basic database publishing and storage • Guestbook	• Email • Group scheduling • Threaded discussions • Tracking applications • Contact management • Full scale custom app development
Provider	Big Fat Web[1] servers (MSN, Yahoo, AOL)	• ISPs • In-house IT functioning as an ISP • Department running single web server	• Primarily IT • Large departments rolling their own infrastructure with big budget and VAR help
Customer	• Individuals • Individuals in LORGs working outside of their IT backbone	• Small Organizations • Workgroups in large organizations that can run their own servers.	Corporate IT and large department IT
Examples	• eRoom • Visto • Netopia • Calendars.net • MagicalDesk • Intranetics • Jump.com • Many more...	• Hway.net • Tripod w/FrontPage extensions • Over 1000 affiliated with the FrontPage Web Presence Provider network	• Lotus Notes • Microsoft Exchange • Novel GroupWise
Office 2000 Integration	None	• File Open/Save • SMTP/POP3 mail • Possible use of threaded discussions / subscriptions for NT hosted WPPs • FrontPage webs including: themes, navigation bars, database publishing	• Outlook as a mail and group scheduling client • Outlook for Forms3 developed forms • Outlook Web Access

In looking at the above table it should be clear that today these three server offerings have significant differences and have unique customers and implementations. More importantly, Office

[1] Term courtesy Bill Gurley, *Above the Crowd*. Fortune Magazine January 24, 1999.

plays an inconsistent role. For Office10 we can focus on providing a unifying view of these servers, at a basic level, and then advancing each experience in a manner consistent with the customers and needs. **Please do not misinterpret this to mean that we will provide an equivalent server experience on each of these platforms** Rather our goal will be to make Office naturally shine in each of these environments.

We will define a basic set of scenarios that will work well across all of the servers we deal with. This will likely include integration with the file open/save dialog and e-mail/Send To, perhaps some features like subscriptions or notifications. For each of these scenarios there will be some amount of "web content" or user-interface design that will facilitate features in the browser. We will design this in a consistent way and there will be one look and feel for any Office customer across these servers.

We are betting on Exchange Platinum and to the degree that Platinum wins over customers in the web provider business or as a server for MSN we are in even better competitive shape. We fully believe that over time Platinum will provide a killer alternative for customers on the web and for ISPs, but even the most optimistic of us feel as though this will take several releases. Some might say we should make an exclusive bet on Platinum, though given the growth of our existing customers (FP/OSE webs) and in the pure web service offerings we would lose out on valuable reach to a broad range of Office customers. At the same time, it is very early in this market so it is easy to see potential convergence or divergence; it is tough to know so we will focus on three core scenarios. We are fully confident that a killer release of Outlook along with Platinum will make for significant inroads versus Notes/Domino.

There is much work to do to research customer needs and define some interesting scenarios. It is to easy to conclude that we need some sort of "project workspace" across all of the servers and to focus our effort on that—but we must be careful to avoid creating this in an inefficient manner by doing it three times (once for Platinum, once for FP/OSE, and once for the MSN customer). A valuable lesson we learned developing Office 2000 was the value in growing the feature pie rather than having everyone focus on the one slice of the pie we already know about. So in this vein I would ask that we look to grow the pie of server features and not implement the same thing multiple times. We will be hardcore about this and it will be a challenge for program management.

Delivering on the promises of Lotus Notes using Microsoft Office and BackOffice

We are losing to Lotus Notes every day in the marketplace. As I write this our most loyal Exchange customer, Boeing, is in the process of accepting proposals for "knowledge management" that is clearly aimed at placing Notes in the corporation. This is over 100,000 seats of Exchange that is at risk, and subsequently over 100,000 seats that could go the way of the Lotus desktop strategy. Even when we win a sale of Exchange to a customer, IBM manages to put us in the position of needing to re-win that sale year after year.

Microsoft has been ineffective in this area for any number of reasons, but one reason that we understand well in Office is that no one has viewed it as their job to compete head-to-head with Notes from the customer perspective. This means feature-for-feature, menu-for-menu, demo-for-demo, review-for-review. With Office10, the Office team is going to assume this responsibility for Microsoft and our customers. This is not something we will be able to accomplish in one release—Lotus as at least a 10 year head start—but it is something we will begin to invest in significantly for Office10.

Lotus' strategy is sometimes hard to understand and it seems to change depending on the customer and their needs, and with each Lotusphere conference. In particular, the element that concerns us the most is the client strategy. Is Lotus trying to have the browser be the universal client and do all the work on the server, or is the browser a quick and dirty or universal access mechanism with all the rich interactions taking place in the Lotus Notes client? Are they really going to push their customers to eSuite for use as "all you really need" applications? Similarly, their solutions strategy has oscillated between integration with the client or providing a separate

tool. In Lotus Notes Release 5, the latest strategy is to have a superset client called the Domino Designer that provides the development tool, though the Lotus Notes client has many elements of the Designer but cleverly hidden away.

Customers have been extraordinarily consistent in their feedback to Microsoft regarding our strategy. They want a simple to understand application for building groupware and knowledge management applications, and they want it to be part of Office. To customers Notes/Domino looks pretty simple:

- **Server**: Domino server supporting multiple protocols (HTTP, SMTP, POP3, NNTP, LDAP) with a straight forward programming model for doing workflow
- **Rich Client**: Lotus Notes that provides a conventional Windows user-interface for doing mail and group scheduling, along with the standard Notes document type that provides the runtime for solutions. It is easy to create new "databases" from this client based on some standard templates. The number of these templates is drastically reduced from the often-overused but rarely understood *nifty-fifty*. The document library, threaded discussion, and issue tracking are key scenarios for this feature.
- **Thin Client**: Internet Explorer or Netscape Navigator for accessing the entire Domino experience from a thin client. Customers get very excited about this, but the realities are quite tricky given the Notes legacy. For example, it isn't possible to simply convert a Notes 4 application to be a browser application. That might be a weakness, but only for existing customers. It isn't clear if new customers are embracing this model and to what degree. This is a super important issue to understand and to deal with appropriately.
- **Tool**: Domino Designer provides a very straightforward tool that exploits the Domino programming model. It is easy to create a new "database" that has easily customized fields, forms, and views, as well as logic. The level of sophistication in this tool is comparable to our VSE/VBE environment in Office 2000. The programming models are, however, fundamentally different.

Our approach will be to provide an even simpler model for customers to understand and to execute on the metaphor even more crisply. We have some incredible assets to leverage in Office 2000 and Exchange and we will do that, but our first order of business is getting in the game on the basics of developing what customers perceive as *Notes apps*. We must not ever lose sight of the fact that we are not even in this game in the minds of customers, so thoughts about out flanking Lotus or providing a suite of competitive products are fine, but only after we provide the simple product that enables the basics of application design.

We will do this by taking advantage of Exchange Platinum and Outlook, as well as our strength in programmability. This is going to be a lot of work, but given the strengths we have it is something we can make major progress on in this release of the ATG wave.

- **Server**: Exchange Platinum provides the rich data store for non-structured information along with a workflow programming model on the server that we can leverage. The server can do all the things we need for this first round, but the client needs to create a programming model around this and provide the full experience. Right now, Platinum is very raw and only provides necessary, but not sufficient, infrastructure. The Exchange team will own making the setup and installation of Exchange as easy as Domino.
- **Rich Client**: Microsoft Outlook is our client. It challenges the Notes client in feature richness and integration with Office, but is behind in some of the Notes Release 5 features. We will do a full competitive analysis and learn from customers, reviews, and analysts about the features that people find compelling. We will create the necessary starter "folders" (aka databases) and customization wizards so that a person starting the Outlook client for the first time can accomplish just as much in the same amount of time as a person starting the Notes client for the first time (against the appropriately configured server).
- **Thin Client**: Internet Explorer will provide a level of richness for accessing data and forms that can't be matched by Lotus' least common denominator approach or use of

applet add-ins. We will carefully balance the customer desire for universal access and we might choose to focus on a least common denominator approach for a majority of applications.

- **Tool**: Microsoft Outlook will host the primary development tool for these applications. Many people have proposed a number of alternate approaches, but all of these add yet another application to the mix (one more than Lotus has). Supporting multiple tools is the right answer in the long term where customers can have a choice (and mix and match), but in the short term we must support the model that Lotus has put forth or we run the risk of continuing to alienate customers with a confusing multi-part strategy. It is worth noting that the fact that the Domino Designer and Notes client are essentially the same product with a slightly modified first-experience. Lotus has chosen to use packaging to make the distinction between the expensive and the inexpensive clients. We will not confuse customers like that since we know the strength of Office has always been "it is in the box". We have incredible strengths in our TCO and UI work that will enable us to offer customized policies for administrators if a customer chooses to hide the tool from the basic Outlook user.

As you can see this strategy is clearly aimed at providing something for customers that they understand based on their understanding of Notes, as makes sense when one is behind. In this first release of an ATG wave we will take advantage of Microsoft's existing and shipping technologies to accomplish this task: Trident forms, Tri-Edit, VBScript, Visual Script Editor, VBA, COM, etc. It is tempting to want to take on the latest and greatest technologies for this release, but we are certain to fail if we first must integrate things that have not yet shipped. Experience has shown that the cost of integrating new infrastructure is itself a process that takes a full release.

From a power-user perspective the scenario might be as follows:

1. Create a new database (folder) by executing File New and choosing "start from scratch" (either on the server or locally) or easily locate and open an existing database.
2. Create a blank form.
3. Add *ad hoc* fields to the form that map to new fields in the folder (thus the user is *designing by form*).
4. Define views on those newly created fields.
5. In the form and in the views, provide a means to write code that manipulates the data at hand (validation, formatting, etc.)
6. Provide a means to invoke richer semantics for the client on the server. This means that from within the code behind a form, a programmer could define or invoke workflow rules that run on the server.
7. Enable browser access to the database, without special casing controls or code.
8. Provide all the necessary development tool support for the "project" management that is equivalent to the Domino Designer view of Forms, Views, Agents, etc. This provides an easy "asset" view of the folder.
9. Support offline views, forms, and synchronization through the rich client.
10. Redistribute the application either on the server or as an easy to use offline store.

In some sense (some might say in every sense) this is a drastic simplification. But it is only with such drastic simplifications that we can measure ourselves since we know this is how customers invariably use the product and more importantly how reviewers evaluate these two products. There is much more to the depth of Lotus Notes and much more we can do as well. A skilled Outlook user would point out that some of these are already possible, albeit not as discoverable as we might need. Perhaps this is not the ideal basic scenario to target, but it is close to the hearts and minds of developers and VARs. We might have to adjust some of the existing concepts in Exchange and Outlook to accomplish this, but that's OK. Lotus is ahead of us and we need to adopt their conventions in key areas if we are to win back some of those customers.

In particular it is pretty easy to jump to the conclusion that we need several "impossibly hard" technologies: fully symmetric client/server programming model, fully unified storage, new forms, etc. We need some hard technologies to be sure, but we simply will not be able to have them all in the first release. We will need to be clever, make some hard choices, and take some lumps from customers. For example, we might define a core set of automation objects that run within Outlook and on the server and provide a *browser-safe* type library to facilitate application development. But any path that has us doing everything in one release is sure to fail as badly as all the past efforts we have had to be in this space.

The Office 2000 infrastructure is an incredible asset when it comes to competing with Notes. For example, Trident forms have already been adopted by Access to connect to structured data. This is an incredibly compelling scenario—imagine creating a loosely structured form that runs in the browser (using the above simplified view of things) that can also add an Office Web Component and have in-place pivot tables connected to SAP data. This is an awesome scenario that leverages both the implementation of this new scenario and the existing Access 2000 product all while Lotus is busy reconciling their legacy forms and web forms. Lotus cannot do that in a 20-minute demo like we could!

For the end-user that starts Outlook in an Exchange Platinum environment, we must also succeed at providing an incredible out of the box experience. The issue tracking and document library template databases in Notes Release 5 are very nice and do a very good job. But one quickly hits a brick wall when it comes time to customize these since their richness in the browser and Notes client hide a very complex set of choices made by the template author. With our use of Trident and the intrinsic power of the VSE engine we should be able to provide templates of equal richness, but with far easier customization. Additionally, Lotus provides some sophisticated inheritance in their relationships between databases, which will be difficult for us to replicate. The three key templates found in Notes 5 include Office Document Library, Discussion Database, and Team Room. There are many others, but few that are useful to end-users.

Finally, we will be very focused in defining key integration scenarios that cross all of the applications. We know we already support file open/save to the Exchange Platinum server. With the PKM add-ins we will have support for the "document management" functionality of setting properties and versioning. We will need to create new scenarios that leverage the power of Office E-Mail and the HTML file formats of our applications. There is an amazing opportunity to do some mind-blowing work in this area, once we nail the basics of getting in the game.

There are three key elements beyond the core Outlook + Platinum work that will enable us to compete broadly with Lotus Notes:

1. **Programmability** We have an incredible strength and depth in programmability. The work driving the integration of script editing in Office 2000 will help us to create great programmable forms using Trident in Office10. We will also investigate using the newer forms technology from development tools, but even without this we are on a solid path to creating a great experience in Office10.
2. **Components**. As Lotus begins to broaden the role of eSuite we are seeing our investment in the web components pay off. Office10 will continue to leverage these components and provide support for key integration with Notes scenarios. The power of being able to combine easily both structured and unstructured data on a single form (in Outlook or IE) is incredibly powerful and surpasses Notes. Data Access Pages provide another unique element for our component strategy as well.
3. **HTML**. All of our applications can participate as full HTML forms which is another key asset. The power of being able to use Excel as the design time for a scriptable solution, or to guarantee the fidelity of documents delivered over the browser without lossy server-side conversion will be a tremendous value to customers.

Competing with Lotus Notes is going to be a great challenge. Notes is a product that has been around for a long time, has a large number of customers, and is backed by a huge sales force. Exchange has proven to be an incredible powerful competitor but to take things to the next level

and surpass customer expectations and solidify the Exchange sale we have to answer to the Notes and Domino products head on.

Killer web scenarios for knowledge workers and IEUs

Clearly our biggest bet in Office 2000 was on the departmental web site. Departmental webs also provide compelling IEU scenarios, though they will be a slow burn due to the issues of deploying web servers. We have generated a lot of excitement with these features and have an incredible foundation upon which to build some higher-level features that solve important IEU problems and scenarios.

Our goal for this focus area is to enable the coolest, easiest to set up, and most productive webs for sharing and collaborating in a department or on a hosted ISP. The key assets we will leverage are the HTML file format in our authoring tools, FrontPage web site authoring and management, and the Office/FrontPage server extensions.

Although there is some obvious potential for overlap between the Notes scenarios and killer workgroup webs, we want to be in a position to continue to provide the best support for straight-forward HTTP servers, which will continue to dominate due to their simplicity of installation and management. Our job is to have directed features for each of the server spaces. Where we do overlap we will consciously choose to do so and will design a consistent user-experience at critical levels. This is challenging because in many ways Platinum is defined to be both a web server and a groupware server (or a groupware server that is a web server). Our goal is that everything we do in this space should just work on a Platinum server—with the support for the installable file system and full IIS this seems like a reasonable goal and one the Platinum team is willing to work towards.

A key goal of the departmental web is to be easy to install and administer. The administrator of a departmental web is often the administrative assistant of a group or the IEU, and significant knowledge of NT server cannot be a prerequisite. Today with FrontPage all of the administration is contained within the FrontPage experience and this absolutely must continue.

There are many elements to the cool IEU web site and there is much research to be done to help come up with a definition. This goes beyond knowledge management and must include the ability to create presence sites (whether in a corporation or on the internet), commerce, and rich database integration sites. This is an important element of growing the pie, rather than focusing on imitating Notes functionality without leveraging Platinum we should define a broad set of scenarios that take full advantage of user-created HTML pages and data.

We will define a fine line between the functionality on a web server using our server extensions and a web service offering on MSN. Ideally we would just host our extensions on MSN, but the scalability and management concerns clearly indicate we should define a subset of features and engineer those for the MSN data center environment. Our job will be to innovate appropriately but to maintain a clear relationship and synergy between these two server environments.

An issue that is worth considering is our investment in cross-platform support. As we know from Office 2000 there was considerable complexity, and downside, to having the complex relationship between server extensions and platforms/servers. Ideally we would offer the full and rich experience on Windows NT and ISPs would just make that a premier offering. Unfortunately many of our customers have standardized on Linux/Apache for their web servers. We should carefully analyze the trade-offs involved in having deeper Linux/Apache support. Our dependency on SQL for the web discussions also showed that we should be careful about any data that customers could perceive should be part of Exchange. For example, discussions, user lists, calendars, etc. are all clearly Exchange data types and customers will want to have Exchange be the source for such data and will want the full power of Exchange for manipulating it.

When it comes to picking "data stores" it is easy to just want to be unified, but given the short time frame of this release and our desire for an easy to administer solution, we should carefully pick

these architectural changes. Any data we move to a new storage strategy must do so with a clear end-user benefit in this release.

The bet we are making is that webs will increasingly become a critical element of access to tasks and information in a corporation. Work will start on webs (probably from a mail message with a URL) and progress through a variety of sites in a corporation. These sites will provide access to all sorts of collaboration scenarios that we can create in Office. We should be careful not to emphasize too much a notion of a "portal" for Office-specific work since there will be many portals (not unlike what we used to say about *home pages* when it seemed that everyone was producing *the* home page). Given that there are likely to be many portals, it behooves us to have a great way to proliferate any portal created with or populated with Office Premium!

That said, a key element of a successful implementation of killer webs involves being able to create a killer "portal" (or *project space)* with easy to use FrontPage along with an easy to setup and maintain server, or an ISP running FrontPage/Office server extensions. We will necessarily create a meaningful project space experience for Office customers, but one that is flexible enough to be broadly applicable in the workplace. Some of the early efforts on TeamPages and the prototype http://office10 site are interesting. We will also need to understand the customers who are using and liking eRoom, HotOffice, and others. Again, a key element of project spaces is that they scales up/down to an experience on MSN's service.

There are a number of other elements to creating killer webs and much research to be done to learn how people are using Office 2000 or thinking about deploying departmental webs. Some examples of features in this area include the following—this is not a complete list, but rather some suggestions.

- **Group authoring**. The Word team has been researching the process of collaborative authoring and what it means for documents to be created by multiple people. This is an excellent area to leverage the web infrastructure.
- **HTML**. Our main challenge with HTML will be to advance the richness of our round-trip document formats while at the same time maintaining compatibility with our Office 2000 files and applications.
- **Site Development and Management**. We have the single best site management tool for the masses in FrontPage 2000 and we should continue to enhance it. There are numerous areas worth improving. This is a good area where some specific integration with PKM could be very beneficial to our larger customers.
- **Annotations**. It is clear from the reception of Web Discussions in Office 2000 that the idea of commenting on web pages resonates with customers. Yet our implementation had limitations—the dependency on the server and lack of cross platform support, and the lack of rich text or graphical annotations tools are two. We should look hard at the scenario of an acetate layer on top of documents/pages and the basic tools (highlight, drawing, stickies). We know this is a compelling scenario when shown to customers, and the challenge is implementing it in a general way.
- **Subscriptions**. Similarly, the subscriptions feature in OSE 2000 is another compelling scenario. We have only scratched the surface of integrating this feature in our process—can you subscribe people while saving the document, can we take advantage of Outlook by sending a special message, etc. Additionally we should look at how this might scale to a web service on MSN.
- **Lists**. The TeamPages work has shown that there is significant flexibility in a basic structured list that is easily created and customized through IE. There is potential in this area as we look to enrich the process. The connection of this feature to our strategic investment in Platinum is something to think through completely.
- **Better meetings and status reports**. A common scenario that is often mentioned is using a web site to make meetings more effective (storing meeting notes, connecting to slides, etc.).

The integration of mail and the web server present some interesting scenarios. For example, being able to mail a page to a web server makes it easy to turn a common post-meeting process to something that is easily added to a web server. Similarly, status reports that can roll up tracking information in a meeting report (action items) and send mail or post a page are very interesting to larger workgroups (see Lotus Team Room for an example of this)

Innovating the customer's Office experience by integrating exciting new technologies

We're in the unique position in Office to be able to take technologies and integrate them into the customer experience in a sensible way, which can help to make new technologies more appealing The challenge for our efforts in this area are to find a consistent set of themes and build a feature set that supports these themes. At this early phase all we can do is list some of the technologies that are making a splash now.

A key element of the new technologies we choose to integrate is the availability of services on the web. Although many of the scenarios might revolve around what we would call a knowledge worker, the integration with web services is distinct because it involves customers that might not be interested in or might not have access to a traditional web server. More importantly, these influential end-users might be executives or others in a large company that use services on a web in addition to being a consumer of information produced by knowledge workers. This opens up a range of possibilities to integrate with our existing MSN services (HotMail, buddy list) and to create new services in collaboration with MSN.

So the theme of "influential end-user" will permeate our efforts in integrating new technologies. IEUs are the folks that have CE devices or more likely Palm PCs, are trying out web services, want to integrate their disparate sources of information (voice mail, fax, email), etc. Our compelling demonstration for power users will likely include a number of these key features and the themes of this area should resonate with IEUs. A major theme of these efforts then will be "Office anywhere" meaning that the key elements of Office are available as appropriate on a variety of devices and through a variety of means.

- **Voice.** Much has been said about the role of speech, but the biggest issue we must overcome is doing a better job of integrating speech technology than our competitors, and we must make it much more useful. We must of course do dictation, but how can we do it better? There are other scenarios that become interesting such as integrating voice mail, adding voice annotations to documents, improving voice narration in PowerPoint, as well as providing the basics of voice editing and touch up.
- **MSN Integration.** Ideally, we would just host our implementation of Platinum on MSN and the full richness of our experience would be available to MSN customers. An alternative would be to host the server extensions on MSN, but that too has scalability and manageability concerns though that will not stop us from improving our server code and making it work for the scale of MSN. Both of those approaches fail to offer the streamlined solutions that customers have access to today. For this release, until we can scale Platinum+server extensions we will design a set of scalable features in concert with the MSN team. The primary goal will be scalability, reliability, simplicity, and integration with Office and existing MSN work (HotMail). We will likely focus on a few key features and make sure they work, above creating a lot of features that might not scale or be reliable enough.
- **Chat and Buddy List.** A new element of MSN's offerings is going to be chat and buddy list integration. It is also going to be a cool feature for Platinum. Integrating this into Outlook could provide a rich experience.
- **Windows CE.** Windows CE is gaining momentum and there are exciting opportunities to do a much better job integrating Office and CE. Customers are very vocal about the problems of synchronizing their data between Outlook and the CE devices, and we can make this much less painful and much more seamless. There are other interesting pieces of data we can synchronize, such as spelling dictionaries, or features we should encourage the CE team to add to the Pocket applications in a way that is consistent with Office. We have an opportunity

to offer customers a rich platform for mobile computing and for non-PC devices and if Office can play a key role in this we can offer some unique features and advantages.

- **Fax**. There is some irony in the mentioning of fax technology as a new technology, but for many customers (especially small business and legal) fax is still a mission critical element of their daily work. Although we've done some work to integrate third party support with Outlook, we can look at creating more of a full experience within Office so that Word and Outlook can work much better together at the fax process, and so that Outlook can work to treat fax information in a more "first class" manner.
- **Conferencing, meetings**. Meetings are a necessary evil of big business that people wish we could make better with Office. PowerPoint has enabled most meetings to move from acetate to online presentations. Many large customers are deploying NetMeeting universally and we've seen members of the OAC who swear by the use of NetMeeting. We've done some integration in Office 2000 and customers appreciate that. A next step for this area is to really introduce a process for really making meetings more of an asset—starting from the idea to schedule a meeting, to creating the agenda and presentation, to notes, and tracking follow-up. We have many of these pieces scattered throughout Office, but we do not have the notion of a "meeting wizard" to tie them all together. There is also another opportunity for integrating CE devices, since more and more the notes resulting from a meeting are being taken on a handheld.
- **XML**. The role of XML as the ASCII format for data is becoming increasingly important to the server side of things. Most of the BackOffice applications will be able to offer up an XML description of data to clients that can understand it. We have an opportunity to hook our unique data access assets to these XML data sources. We must be careful about our file format challenges, but the possibilities of leveraging the huge amount of work going on in this area are endless and put Office in an important spotlight.
- **Components**. We made a significant investment in components in Office 2000 and customers are quite optimistic about them. One set of customers that we did not originally target is super excited about them—the Enterprise Resource Planning software products (SAP, for example). How can we take these components to the next level and allow them to communicate at a rich level with SAP? What are the sorts of scenarios that we should create new components for? How can we improve the role of components in our applications so that more users can create component-based web pages? There is much to learn as we research how our Office 2000 components are used by customers.
- **Graphics and animation**. We have an opportunity to raise the bar on graphics and animation in the document creation process by taking advantage of PhotoDraw and Liquid Motion (VizAct). We know IEUs value the "coolness" factor of documents so this is potentially a great way to get customers to jump up and say, "I want to buy Office10".

Nailing the fundamentals by making Office better at its core

All the new features in Office 2000 won't matter at all unless we can make substantial progress on improving the performance, reliability, stability, and usability of the core scenarios of the core applications. We have heard time and time again that people would like a release of Office that didn't add any new features, but just fixed the bugs. We know that customers won't really buy such a release, so our challenge is to find ways to re-engineer core scenarios, while improving the usability and richness of those features. This is easily the most important challenge for Office 2000—it is more important than beating Notes since it impacts 100 percent of our customers.

The focus of this area needs to be a set of investments that accomplish two things:

(a) Substantially improve the average Office customer's experience in using the product
(b) Turn the crank one more time on the key investments we made in Office 2000

In terms of improving the basic scenarios, there is much we can do. Primarily the feedback we get from customers is that Office is getting unstable each release and the challenges for a customer (or a PSS engineer) in tracking down a crash or instability are enormous. How can we make this whole area of Office much improved, in a consistent way across the entire box?

- **Never crash.** Office should never crash in our own code. We need to gracefully handle any boundary conditions that might result in an inability to save a file or recover from an unexpected catastrophe. PowerPoint has the most advanced error handling to date and it is worth looking at improving this and building it into the whole product. When Office does crash we need to do a better job at reporting the state and possible diagnostic information. This is a very deep area and there has already been some early work on this posted to the Office10 web site.
- **Better error handling.** We've heard from PSS that Office 97's error messages and error handling are good for the basic cases, but it is too easy for customers to get in actionable error messages or for PSS to get stuck unable to identify the source and reason for a message. Again, this is a deep area with many potential fixes that we should explore. Administrators are interested in having a more detailed set of errors logged to the Windows 2000 event log—for example, we could choose to log failed add-ins, crashes, corrupt files, etc. The ability to turn on this logging might slow down the application, but it also might provide needed diagnostic assistance to support personnel.
- **Safe mode**. We have an ad hoc collection of safety features in our applications today and customers want more. What can we do to offer a "safe mode" for Office that will allow the applications to be used in the event of a setup failure?
- **Feedback to Microsoft.** Office 2000 has a very nice "complain to Microsoft" feature. How can we build on this feature to explore having both a closer relationship with customers but also much more of a feedback loop regarding product problems, failures, and suggestions?
- **Customer satisfaction**. We need to be obsessive about drilling into the details of PSS calls and wish list requests. We need to do our part on the product team to improve the satisfaction with our product.

For every effort we made in Office 2000 there is much work "left to be done". The challenge with rushing to improve what we are about to ship is soliciting the right amount of feedback on Office 2000 so we are acting in a manner consistent with how customers are using Office 2000.

- **Address top concerns**. We will be inundated with QFEs and DCRs for Office 2000. We will execute many of these, but some will be invasive enough that we will choose to postpone them until Office10. We need to be very proactive in learning about these features and implementing them in Office10.
- **Reduce TCO to zero**. The investments in TCO were awesome in Office 2000. Administrators love the features. We will certainly have some issues to address, and most will be addressed in QFEs. We need to take this to the next level of simplicity and ease of use. In particular, emulating the Macintosh Office 98 drag and drop setup and resiliency is something that end-users and corporate customers will likely respond to very positively.
- **HTML**. It goes without saying that our HTML support in Office 2000 is an unparalleled accomplishment. It is still early in the lifecycle of this investment and sometimes that shows. We know about the challenges with data interchange and some of the fidelity issues with different browsing technology. And of course there is an evolution of HTML and new features to take advantage of.
- **Worldwide**. Again, our investments in worldwide support for Office 2000 were incredible and the results blow people away. This is another area where there is still much potential. We have a solid lead in multi-lingual documents, which we can build into an enormous asset.
- **Security**. We have a very nice set of enhancements to make Office appear more secure for customers. We should expect the malicious users of the world to continue to innovate in their space, so we will also need to do the work to stay one step ahead of them.

Finally, each of the core applications in Office has specific investments we should look at in order to continue to innovate in the category and some scenarios that are application specific continue to need refinement. Word still has significant work for the legal community. Excel's largest customers are clamoring for a massive recalc engine. PowerPoint's corporate customers request richer masters for standardization purposes. Access' integration with SQL must continue to be

refined. Outlook customers continue to push the limits of contact management. FrontPage ISPs want more hooks and features for customization, commerce, and other ways to provide moneymaking services for their own business.

Competitive Issues

Our competition in the marketplace is as fierce as ever. Rather than dive into the specifics of what could be over a dozen products, it is probably a good idea to think about the basic categories of competitors. We will need to keep this in mind as we begin planning Office10. In some ways we were not as focused on specific competitors when we began Office 2000, but rather were focused on broad industry trends. This time around we've got some specific products and services to concern ourselves with.

Competitive Space	Concern	Products
Existing Office	Creating a reason to upgrade and providing a no-brainer upgrade that maintains 100% compatibility and does not require a hardware upgrade.	Office 97 and Office 2000
Traditional Competitors	Currently rewriting the review criteria by including speech recognition. Lotus Smart Suite is releasing soon with increased Office 97 (and 2000) file format compatibility. The potential for tighter integration with Notes/Domino could cause us to look at adding features to Office 2000 to remain competitive.	Lotus SmartSuite and Corel WordPerfect. Internationally, Star and Ichitaro.
Component Applications	The role of small component applications as specific competitors to Office is probably more credible this release that with Office 2000. Customers are done being enamored with Java and Java has stabilized to the point that it is possible to build *something* that looks like a useful application. The credible threat is to incorporate component applications with a substantial groupware product. There is a continued threat that these component applications are easier to manage and have lower TCO.	eSuite
Groupware	Never before has the threat been so great to Office. The value of email and collaboration for our large customers as surpassed the value of document creation (from the IT perspective). The "ROI" for Office does not compare with the "ROI" for groupware applications. In addition, the application model for groupware applications makes the traditional "desktop" a liability, rather than an asset as thin-client thinking predominates.	Lotus Notes is it baby!
Web Services	Web services that provide free "project spaces" all make it easy to exchange Office documents. But they all view Office integration as a necessary evil not an asset. It is not hard to imagine any one of them providing basic document creation through components as an option.	ERoom, MagicalDesk, Visto, and a hundred others.
Hosted Office	Although not a competitor directly, the risk that Office will become a hosted server (i.e. over Terminal Server) means that fewer IEUs will have their own local copies of Office and thus their ability to upgrade their machine on their own.	Office 2000 is likely to be the preferred hosted release.