

---

**Subject:** Building on Windows 7  
**To:** Windows Product Team and Development Partners  
**Date:** May 4, 2009

---

We hear these words often—*Microsoft Windows is the most widely used software in the world and one of the most successful businesses of all time*. We must not lose sight of the depth of the responsibility and the humility with which we will approach the work of planning and building a new release of Windows. In the technology industry success at any scale is supremely hard to achieve, and maintaining success can only come through clarity of purpose and excellence in execution, and some good fortune. Our purpose is to build and sell the most popular software platform used by individuals, organizations, developers, and to do so while doing our part to participate in and contribute to the vast PC ecosystem. Our execution in all aspects must continue to improve and remain state of the art. This memo is about building on the foundation established with Windows 7 and transitioning to the next release. While the specifics herein are focused on Windows client and base OS, we are building a coordinated plan across Server, Internet Explorer and Windows Live and many other teams within Microsoft. Windows Server will drive a coordinated plan for the overall Server product and business, building on the planning process for the base OS and client described here, following the timeline here as well and outlined in a separate document. This memo provides the framework we will use to begin to plan the specifics of the release and will do so by offering an overview of the current state of the team, business, product, and then providing the framework for how we will execute on planning.

While everyone is working super hard and most definitely focused on finishing Windows 7, we are all anxious to know “what comes next” across many dimensions—launching Windows 7, adopting new technologies, tuning the organization, and improving our engineering. There is a lot to want to know. At the same time, developing robust answers to these many questions is, as we have done all along, going to be the result of a deliberate and thoughtful process that includes time for everyone to contribute points of view, for points of collaboration to be established across teams, and for providing direction from senior management. It important that we use the time we have between now and starting the development schedule for the next release to develop a shared view of success and excellent plans for our next release.

This memo lays out the direction we will head for the next release, but does not represent the definitive plan nor does it set any “priority 0” or must-have features. We have talked many times about empowerment and accountability—this memo begins the process by which our team is empowered to develop plans while also developing the shared accountabilities for the release. While some may feel like work for the next release has already started, we really begin the journey of the next release of Windows starting today. Everything we learned so far through quests, incubations, platform initiatives, and areas of investigation will be shared as learning for our collective understanding, without any notion of pre-committed decisions or pre-booked work. We’re all approaching the starting line at the same time and will together develop plans, organize, and execute to deliver our next release of Windows.

**Welcome to Windows 8!**

## Microsoft – Windows Team Only

While during the course of building and launching Windows we work very hard to make sure everyone has a clear view of their specific goals and commitments, it is important to recognize that Microsoft Windows is an enormously complex undertaking. The implication is that there is a lot more going on and a lot of background information that is worth absorbing before asking “what’s my part”. Hopefully, by taking this step back each product cycle there is an opportunity for everyone to spend time thinking about the very big picture that is Windows. Below you will see that many times questions are being asked rather than answered, or suggestions rather than decisions are being made. This is by design and should not frustrate you. We’re using this step in the planning process to expand our view of what is possible, what the industry is doing all around us, and what customers are experiencing. Many might prefer much more definitive answers for “what’s next” in the form of “go-no go” decisions on some concrete areas, but as we have said many times the plans come from the team by working together. Others might prefer naming a “point person” for each of the areas discussed. Our approach is far more collaborative and leadership is expected from each and every one of you. With this context and approach we can do a better job planning.

The sections of this memo on Windows 7, Business Opportunities, Competitive Challenges, Customers, Developers and Partners, and Industry Trends each provide background and context. With that context we will outline how we will we are Defining Success, view Evolving the PC, and how we will Structure Windows 8. The final section will provide the timeline for Next Steps.

*As we design, develop, and bring to market Windows 8, we must adhere to the [Windows Principles](#), which as a team we have committed to doing. More broadly, we have a responsibility to operate our business in a manner consistent with the laws and regulations in every market.*

## Contents

Windows 7 .....	2
Our Business Opportunities .....	7
Our Competitive Challenges .....	10
Our Customers .....	14
Developers and Industry Partners .....	17
Industry Trends .....	21
Defining Success .....	24
Evolving the PC .....	26
Structuring Windows 8 .....	29
Next Steps and Timeline .....	38

## Windows 7

As you read this, our team is just starting to see (legitimate) downloads and installations of our Windows 7 Release Candidate. Congratulations! When we look back at where we were as a team in the summer of 2007 as we worked through the planning phase for Windows 7 and developed the *Windows 7 Vision*,

## Microsoft – Windows Team Only

we can see we have come a long way—we've come a long way with the code, a long way as an organization, and a long way in terms of how we get things done. It is definitely too early to declare success in all dimensions, but we can certainly say with confidence that Windows 7 is well-poised to be a fantastic product; our organization is healthier and far more accountable; and as a team we have a clearer sense of roles and responsibilities across all the disciplines that contribute to Windows 7. These are no small accomplishments and everyone has worked incredibly hard to take us from Vision to Release Candidate so it is worth calling out some of the specifics.

Windows 7 has been extremely well-received by customers. Estimates are that we have had almost 3 million PCs using the Beta and thus we've had the highest degree of meaningful engagement in improving the product through the beta process that we've ever had. By far the most exciting aspect of the work that got us to where we are today has been the strong focus on delivering external builds that are high quality (in all dimensions) and usable on a daily basis by engaged community members. Starting with the Pre-Beta at the PDC and then adding the Beta in January, each step has demonstrated that we can have a focused plan, deliver a milestone release that meets our exit criteria, and then have customers (internal and external) experience the product the way we believe it should be experienced.

We made significant enhancements to how we engineer Windows in order to accomplish the above results. Starting with the Windows 7 planning process and continuing through improvements like reliable daily builds, improved test automation, improved milestone discipline to cut features and scenarios that were not ready, preparing early for servicing, strong incorporation of Windows design in the UX, improved telemetry, and many, many others, teams across all of Windows improved how they worked both within and across teams. The general tenet of frontloading work of all forms—engineering system, design, architecture, integration, test plans, and so on—has proven to be enormously helpful in engineering Windows.

The positive reception from all those in the community that experienced the release has been extraordinary, rewarding, and humbling. As difficult as it was to work through the experience of Windows Vista in the market (one which has dramatically improved since RTM), we have seen customers shift their thinking to Windows 7 in an unprecedented way. Like a long-lost friend, Windows loyalists, PC enthusiasts, and our industry partners have shown that if we can predictably deliver an exciting release of Windows they will match that excitement with their own creativity, energy, support and enthusiasm. While we've circulated the press reports on Windows 7 Beta reviews this is also a good time to remind ourselves of some of the positive reactions. It is worth noting that the stability of the product and the interest in the product created an unprecedented wave of reviews never really seen before on an unreleased operating system:

- “This first beta of Windows 7 is a polished piece of work, with few apparent kinks to be worked out. Windows 7 is much further along at this beta stage than Windows Vista was at a similar point.” *Preston Gralla/Computerworld*
- “Microsoft's next OS is looking like Vista done right, with a smaller disk and memory footprint, faster startup, and fewer annoyances. We shouldn't see the kind of hardware incompatibilities encountered with the first version of Vista, since 7 uses much of the kernel code from the

## Microsoft – Windows Team Only

previous OS. Still, users who cling to XP will have a bigger adjustment interface-wise than those who moved to Vista.” *Michael Muchmore/PC Magazine*

- “As for performance, I think Microsoft is going to surprise people. And if you're working for an enterprise that hasn't upgraded to Vista because it won't run acceptably on your existing PCs, you'll want to look again at Windows 7.” Paul Thurrott/Supersite for Windows
- “I'm actually having fun using Seven, something I haven't said about a Microsoft operating system since Windows 95.” *Joe Wilcox/MicrosoftWatch*
- “Windows 7 seems to be much improved than Windows Vista in lots of areas and the beta clearly shows that Microsoft has indeed taken users feedback on Vista seriously.” Chandran *Chakkaradeep/ Neowin*
- “Microsoft has taken a disciplined approach to planning, building and delivering Windows 7 (and planning is already underway for Windows 8); this may lack the flamboyance of previous versions but if Microsoft can maintain the quality of the beta, Windows 7 could indeed be the best version of Windows yet.” *Mary Branscomb/Financial Times*
- “In general, I have found Windows 7 a pleasure to use. There are a few drawbacks, but my preliminary verdict on Windows 7 is positive....Compatibility with hardware and software, which was a problem in Vista, seems far better in Windows 7 — even in the beta. I tried a wide variety of hardware, including printers, Web cams, external hard disks and cameras, and nearly all worked fine.” *Walt Mossberg/Wall Street Journal*

These are just a few of the quotes from hundreds of unique reviews with an overwhelmingly positive tone across the board. With the RC and as we work deliberately towards RTM and *General Availability*, we are well-positioned for Windows 7 to have the most positive reception, perhaps since Windows 95. Finishing Windows 7 with the same level of focus that got us this far is of course the key—it is up to us to complete the release with the excellence that got us this far.

Yet the common theme has been *Windows 7 as it compares to Vista*. This comparison is not going to happen again and as we think about Windows 8, we have to keep in mind that we have, as a team, most decidedly moved the baseline for future comparisons. We have raised the bar. Thus along with the positive, we must, as we transition to Windows 8, begin to look at some areas we will need to improve as a team. Many improvements we must will make build on the improvements we have made in building and marketing Windows 7, but as a learning organization we must never lose sight of the fact that we cannot ever be fully done, and by definition there is always more room to improve and more to learn including the following:

**Planning and engaging with partners.** Windows is a product that cannot stand alone. Windows is only as good as the programs people run, the computers people use, the hardware and peripherals people choose, and the deployment and management tools. While we have built on the foundations of many years of work, including a significant transition in many areas such as graphics, the reality is we are still not where we need to be in working with partners. The most important partners we need to re-engage with and re-ignite are developers. We've been unclear and confusing with developers of all types—ISVs, enterprise, and web developers. Windows 8 will be an opportunity for us to clarify our message as we

## Microsoft – Windows Team Only

take deliberate steps to align our efforts with a release of our premier development tools that synchronizes with Windows 8. Our OEM partners are undergoing significant challenges in their business as opportunities for differentiation, competition, and challenges of lower priced, but adequately powered, machines enter the picture. With the development of Windows 7, we took important steps to rebuild our trust in product development by engaging early, but also reliably and honestly. Windows 8 is a chance to build on that increased level of product development trust and broaden it throughout the OEM organization. IHVs have been stretched thin by the transition to Vista and the transition to 64-bits. Our ability to effectively partner across many device classes and businesses in hardware has been more difficult for many of the same reasons experienced by OEMs. Windows 8, building on the foundation of a very successful series of Windows 7 events such as WinHEC and the DDC, will need to take new steps to more reliably work with partners and most of all improve how we surface the improvements in the hardware/software boundary to customers through Windows Update. Our learning from Windows 7 tells us that we must think hard about how we enable the financial success of our partners through the APIs, features, and programs we create.

**Building scenarios across feature teams.** Windows 7 saw an unprecedented level of collaboration across teams and we have many features that show when we have shared goals and a clear view of where the product is going we can deliver the best of “plumbing” and “experience”. Features like HomeGroup, Device Stage, Action Center, Multi-Touch in Media Center, IE, and the work we did on UAC demonstrate that we have a team willing to work across feature teams and get well scoped work done in a single release. We need to move to a model where we default to cross-team work and don’t view it as the exception to squeeze into the schedule. Perhaps we need to focus on fewer scenarios relative to Windows 7 that are more detailed upfront, and delivered in a more comprehensive way. We need to make sure that we build with clear boundaries that make sense to customers and not default to plumbing and UI, front-end and back-end, or other separations that emphasize organization, architecture, or the code. Great software is great to end-users and developers when it completes a scenario end-to-end, whether in features or API support. Great engineers can work across these paradigms from design to build to test to delivery.

**Sharing more.** Windows 7 was a much improved product cycle in terms of shared flow of information and shared processes. We have much work to continue to do on our engineering system to bring more consistency, and with that more flexibility, predictability, and efficiency. An area where we have to make significant improvements in Windows 7 is in how we share code across the operating system and how we deliver on an OS experience for admins, developers, and end-users that looks more consistent, uses the code that we evangelize, and is more seamlessly managed. We can’t rewrite all of Windows to use all the shared code, but we are going to look clearly at areas where we are sharing some but not enough, whether that is code and/or process. Setup, management, UI frameworks, SKUs, control panels, policies, registry usage, controls, and more are examples of where we need a deliberate focus on improving our customer experience by improving our engineering through sharing code and process.

**Developing for business flexibility.** As a business, Windows is one of the largest product businesses in the world. As we have said many times, we cannot sell \$15 billion dollars of software with one price, one SKU, and one business model all around the world. As much as the PC enthusiast in all of us might

## Microsoft – Windows Team Only

like “one SKU at one low price” for Windows, that is not a broadly customer focused approach nor is it practical at a global scale across so many different customer needs. Engineering for the flexibility we need in the business is an enormous challenge because we can’t always know what business challenges we will face in real time and what *knobs and dials* we might like the opportunity to tweak to respond. In the challenging market we face we will need to be deliberate in planning Windows 8 to provide new tools for business flexibility that are built into the product, easily changed without changing code, and manageable on a worldwide licensing basis.

**Engineering for performance.** Windows 7 will be the first release of Windows not to double the system requirements from release to release—yes, you read that correctly. While each new release of Windows has provided significant new functionality and capabilities, the cost of “upgrade” has consistently been viewed as a negative, whether that is because of system requirements which drive poor performance on existing hardware, or OS support for existing devices and hardware. With Windows 7, as a team we proved we can provide a broad array of new functionality while at the same time we focused clearly on performance without looking to clever engineering to mask or minimize the clear challenges inherent in the code. We focused on the raw performance of hundreds of benchmarks throughout the development cycle and delivered improved memory, CPU, and I/O improvements consistently across many subsystems. This is a vast improvement as a team and substantially changed the landscape for the business! Windows 8 needs to have at least another round of significant improvements without increasing system requirements. That does not mean we will not bet on new hardware innovation, but rather not relying on it for the baseline experience or even better finding ways to incorporate it into the baseline and still keep requirements constant. In fact, as a challenge to the team we should see if we could reduce the system requirements for Windows 8 or at the very least running on our benchmark hardware we should see improvements in processes, threads, handles, I/O, CPU, memory, and so on. As we talk about the evolution of hardware we will see why devices with low specs and 32-bit processors will continue to be immensely important. Equally important in this dimension and perhaps more so as we work through planning, will be the disk footprint of the product overall. We believe we can and should surprise the industry again with the performance of Windows! On top of the challenge to run faster than Windows 7 with a clean image, we want Windows 8 to stay performant over time as customers, both consumers and IT Pros, install applications and drivers. There is an industry wide derogatory term for this performance degradation, *Win-rot*. We don’t like this and we know we Windows 7’s new performance telemetry can show us real ways to keep people’s machines fast.

**Engineering for quality.** Everything that we would apply to performance we would apply to quality as well—Windows 7 was a significant improvement in overall system quality from pre-Beta to Beta to RC. We can do even better. From how we move quality upstream to the automation coverage and to our efforts at preparing for sustaining engineering each of these and more are areas where we can and must improve substantially in Windows 8. Perhaps more importantly is to continue to develop our “muscles” for accountability and decision making that make it easy for us to make sure that decisions about quality are made as close to the team that owns the code as possible, in real time, and that those decisions are best made there and made best there.

**Engineering for compatibility.** Windows 7 has delivered the best release over release compatibility for both devices and software ever. Even with this accomplishment we know there are too many times where we made changes with unclear customer value that we had to rectify late in the engineering cycle. We can do better on this dimension. Compatibility is a key customer requirement and we must carefully balance compatibility with new features, without freezing the system or merely resorting to duplicate side-by-side stacks of old and new. It is exciting that we will have the best picture of what software is running around the world and how that software is running by using Windows 7's application telemetry. We can use this knowledge to deliver an even better compatibility experience.

**Executing with efficiency.** When we began Windows 7 we also renewed our focus as a team in being efficient—efficient in how we use our schedule time for planning, development, integration, and servicing the final product. We increased the use of tools such as OCA and SQM to be more efficient in how we use feedback to drive changes in the product as we work through milestones—we cannot underestimate the importance of making decisions with real data to support conclusions. And finally we looked at our management structure and operations in terms of the number of layers of management, the number of people involved to make decisions, the metrics tracked, and the “homework” asked of our leads and managers. We demonstrated that we can make the process of building Windows more fun and more efficient and predictable at the same time. As with quality and performance, as a learning organization we need to keep learning about how to “do more with less”. How can we be more efficient in the use of the calendar, use of resources like labs and hardware, managing our processes and team? As we start Windows 8 we will be deliberate in how we improve the process of engineering by first taking a step back and learning broadly through a series of post-mortem efforts that will help us to make sure we find the best list of improvements in overall efficiency.

## Our Business Opportunities

As we begin to plan Windows 8 it is good for us to share a common business context. The Windows business is enormously complex as it participates in a vast multi-party ecosystem. Each party of the ecosystem benefits from many other elements and through some lenses also competes with others. As we have talked about, selling hundreds of millions of units of a product in nearly 200 countries around the world only makes this even more complex. Of course the benefits to executing well on a solid business plan are considerable and with Windows 7 we took the opportunity to deliver on the business goals of Windows with a fresh perspective. For instance, by improving the SKU lineup as we respond to customer feedback, proactively managing through a complex pricing environment created by new low-cost PCs, and focusing clearly on the significant upside to the enterprise business we have put ourselves on solid footing as we transition to launching Windows 7.

At the same time we face a number of challenges which afford us opportunities to significantly and positively impact the Windows business for Windows 8. It is important to look at some of the more significant challenges and opportunities we have as we kick off the planning of Windows 8:

- **PC Sales.** Without going into the numbers, one does not need to look far to see the current macro-economic situation around the world as having a significant negative impact on the sales

of PCs. We are facing a year over year decline in PC sales not seen in almost a decade. The challenges to the Windows business are clear in that we depend significantly on the “run-rate” of new PCs. If consumers do not buy that extra PC or small businesses are not expanding and don’t need a new desktop or enterprises extend the lifecycle of a PC, then the Windows business will reflect those missed opportunities. As a business that is a component dependent on the “run-rate” of a larger business, we must develop our product features (and business models) such that we strive to influence that run-rate measurably, especially in the long run as the global economy recovers. If we make Windows exciting and necessary for customers so that the value is clear and we are able to communicate that effectively then we can influence these macro issues. One way we might think of this is how we evolve Windows to support an *appliance model*—that is a consumer-oriented approach that provides a PC that allows for vendor lockdown, while also providing a high degree of consumer “protection” or a more narrow range of functionality. All of this boils down to the question of can we increase the addressable market for new PCs whether that is by making Windows more exciting, PCs more affordable, or creating new ways additional (multiple) PCs can be used, or other ideas?

- **Premium Products.** The primary way we measure the success we are having with Windows when we are not directly responsible for the run-rate, is to offer customers several different choices for different version of Windows at different prices depending on needs and to then optimize the overall sales mix. With Windows 7 we improved the clarity with which we make these offers around the world. With Windows 8 we will continue to face challenges in the lower overall price for PCs and questions from OEMs, partners, and customers over the “viability” of a premium version of Windows on “good enough” hardware. Do we need a new model to differentiate price points? Whether we do or do not, we need an approach that allows us to tap into the reality that not every customer wants “good enough” and that many customers in all segments are more than happy to pay for an upside in any number of dimensions.
- **Piracy.** Around the world the rate of usage of illegal installations of Windows continues to increase. In some markets this has gotten to such an extreme level that even Genuine Advantage service provided by Windows Update (WU) has been copied so that it runs on third party servers effectively re-creating WU. At the same time, some believe our stance on WGA has caused an even deeper backlash among legal and illegal users of Windows, with particularly harsh feedback in some markets over the past year. This problem has proven enormously challenging throughout the history of Windows (in fact all types of intellectual property) but we are going to continue to put creative energy behind how we might increase the legal ownership of Windows. We recognize that we must strike a balance between usability and systems management in doing so. For Windows 8, how can we look at dramatically different approaches beyond the OEM model for key markets such as China and Brazil? During planning we can begin to develop an approach and connect that to our overall needs to engineer for business flexibility.
- **Emerging Markets.** It might go without saying, but we will certainly make Windows great in every country and in every language for which we choose to build Windows. Emerging markets, by their very definition, present us with the biggest opportunity and biggest challenge. These



markets, because of their size (India, China, and Brazil in particular), the run-rate of PCs, and the sheer number of people that experience computing with a cell phone or PCs in internet cafes, are significant opportunities. Yet as we know these markets also exhibit our highest piracy rates and are generally viewed as the most price sensitive, particularly within the market. We've tried many different approaches in the past and as with piracy, this is an area we will continue to spend our creative energy on new products and new go-to-market strategies. We need to look deeper at unique needs for these markets, not necessarily adapting what we develop for the rich, but rather translating each market's value system into solutions that are uniquely relevant. This is an area we can learn from related businesses or local competitors, especially in markets like China. Simply being low priced or pushing anti-piracy will not be enough. As we consider the emerging markets of the world, we will likely learn that each market has unique needs and opportunities and so there will necessarily be some tradeoffs involved. We are going to be especially aggressive in how we consider new business models and changes in the way we operate when it comes to achieving success in China. A key part of what we need to accomplish is integrating the choices we make for the business in the planning process up front so we can engineer the product to support these aspirations.

- **Windows Live.** *The Rookies* commercials began a vigorous campaign that shows that the value of Windows increases when a customer uses Windows Live. The demonstration of Windows Live Photo Gallery in these ads connects with potential customers and also shows the connection between Windows and Windows Live that is so important to our business. We will of course continue to connect Windows Live to a variety of third-party services, work with a variety of browsers, and use a broad array of open protocols. At the same time we will use the published APIs of Windows to deliver a premium client and service experience where we think we best demonstrate the value of Windows Live. As we plan Windows Live Wave 4 (and beyond) we will be very deliberate about how we build on the latest Windows client proposition and APIs and showcase the value of Windows and Windows Live. The other part of this equation is doing an amazing release of Windows Live that has leading features, compelling scenarios, and world-class scale and performance. Wave 4 is currently being planned with these goals in mind.
- **New Business Models.** While the primary business of Windows is through OEMs, we have a very significant enterprise volume license component as well (and must continue to reinforce the enterprise value proposition). This royalty model has remained relatively "unchanged" for a very long time. Any time a model remains relatively static the door opens for alternate approaches as customer needs change or as partners look to gain an advantage over their competitors. Many have tried alternating PC models in the past such as advertising supported, annuity, pre-paid, and even using free operating systems. The "bounty" model of paying for placement of software on a PC continues to be very popular with OEMs and so it is natural for it to begin to include more and more software as long as companies with that software are willing to pay. While none of these have yet to substantially change the way PCs are sold with Windows, there is a newfound energy around alternate models. We need to continue our explorations of alternative models, especially as they enable our ecosystem and distribution channels to sell more Windows, for more people, and across more devices, worldwide.

- **Extending PC sales channels.** Aside from the innovation in mobile communications on a technical level, the mobile communication industry has also pioneered a model based on subsidizing hardware over the lifetime of a communications subscription (to lower the cost of becoming a customer). For PCs this model is becoming attractive and not only coincides with a rise in 3G phones for internet access, but also the increase in usage of WWAN configured PCs and the success of the iPhone (which itself upended the traditional business model of mobile phones). In every country the mobile phone carriers are the largest advertisers with an enormous number of points of sale outlets. There is a strong desire to sell PCs (that is devices that can access the high speed wireless network to connect to services) through these outlets. To do so requires a new subsidy model for both the PC and Windows that fits within the cash-flow view of the mobile companies that is distinctly different than our OEM royalty. We also need to understand operational implications around deployment, sales and support. Within this are significant product needs as well such as a “consumer electronics” view of “reset” and “lockdown”, smart network management, roaming behaviors, and many more that we should investigate. Windows 8 will offer us a chance to be clear about how we will engineer our product and design our business for this new customer.

As we can see, despite the broad success of Windows in the market, we have many challenges. This list is not complete as we face significant business challenges in each of our traditional customer segments—enterprise value, consumer excitement, small business functionality, and so on. At the same time, we also have significant opportunities to broaden our market beyond traditional channel and business models. As we plan Windows 8 we will be clear up front about how we, through our business team, will develop plans for significant business opportunities and see those plans through the execution of the product starting at the Vision step. As part of Windows 8 we are going to be more deliberate and planful about choosing which business opportunities to tackle, developing complete business and product plans up front for these opportunities, and making sure we have clear points of business flexibility engineered in the product. Because we are part of a creative and diverse ecosystem we cannot plan on solving for every potential opportunity thus the planning efforts for this area need to be more focused, clear, and consistent and combined with better follow-through than in past releases.

## **Our Competitive Challenges**

At every step in the evolution of Windows, the product has faced strong competition. While we can look at the relative sales or installed base of various products, we are always going to choose to pay attention to the strength of the technology and accompanying business model. Regardless of the measure of success, we know that sustained success in the technology world requires us to keep an eye on what is catching up to us as well as what technologies or technology products might “disrupt” the technology and product advantages afforded by Windows. Windows 8 is no different and we have a broad array of competitive products that we need to take into account as we work to win customers in segments and markets around the world.

## Microsoft – Windows Team Only

No understanding of the competition would be complete without first talking about Windows XP. The installed base of PCs running Windows XP will continue to be an opportunity for us. With Windows 7 we made a smart choice in not “upgrading” these PCs in favor of a much better customer experience. Our goal for Windows 8 is to make a release of Windows that is compelling enough for customers to acquire new PCs, replacing those (now aging) Windows XP PCs, or to enable a clean install on relatively older hardware because we will not increase our system requirements. This is no small challenge. We have faced two significant challenges relative to this goal. First, because of the long time between releases and early reception of Windows Vista, many look back at the release of Windows XP with (excessively) rose-colored glasses despite early reviews and receptions that were fairly lukewarm at best (we could also learn how time can change perceptions and how early initial reviews might not play out over time). Second, because of the system requirements of Vista and the release of lower-capability CPUs with limited storage we saw a gap between available (and desirable) hardware and the system requirements for Vista, particularly in disk space. In order to make sure customers had the option for a Windows product on this hardware we chose to continue to sell Windows XP even in the midst of Windows Vista. We should learn the lessons on the importance of a sustainable and predictable release cycle and the need to maintain reasonable system requirements, application and driver compatibility release-over-release.

Our competition covers a broad range of products and technologies and for this memo we will focus on the client operating system and experience. We face competition from traditional Operating Systems in Apple OS X and Linux.

- **Apple.** Apple OS X is a very strong product coupled with amazing hardware. The transition Apple has made from OS 9 to a modern OS architecture on Intel hardware is on par with the transition we made to both the NT code base and to 64-bits. From an OS technology perspective we see the strength of OS X among universities and administrators who find the Mac (Mach-based) kernel and associated shell, tools and techniques “comforting”. From a user experience perspective, we cannot be defensive about the reality that Mac hardware + OS X + iLife continues to be the standard by which a PC + Windows 7 + Windows Live will be judged in terms of technology, and then the purchase experience, post-sales experience, and ecosystem have grown to be considerable strengths. While we have only some details, the hygiene work being done on *Snow Leopard* is likely to generate significant positive views as the OS becomes “leaner and more streamlined” and likely claims about being more modern with respect to graphics, 64-bits, and user-interface. As we describe below, the sharing of code and architecture, particularly for important strategic elements of the OS, between the iPhone OS and Leopard is technically interesting and certainly responsible for some elements of the platform success. Apple is not without blemishes, but in planning Windows 8 we must focus on their strengths and assume they will continue to execute well on those. Being deliberate and informed about competing with Snow Leopard and, relative to iPhone, making sure we build on the assets of Windows for our Windows Phones should be strongly considered.
- **Linux.** Like so many competitors we have faced as a technology company, just as we thought “this one won’t reach critical mass” we saw two events provide a breath of life into “Linux on

the desktop”. First, the rise of **Ubuntu**, both the technology/packaging and the “movement”, has created a rallying cry for OEMs and for the Linux community. Second, the low-cost PCs that initially came with Linux (due to the footprint “requirements”) created a new incentive for OEMs/ODMs to find a way to make it work. While we were effective in establishing a value proposition for Windows XP on these PCs, the seed that was planted will continue to be revisited by PC makers and designers. They appreciate the potential for business model innovation, componentization, tailoring, and also the opportunities to differentiate using the open source aspects of the OS. It is worth noting the reality that Linux on the server continues to dominate in many important workloads and the server plans are going to inform the planning of the base OS work such that we are extremely focused on defining specific scenarios where we make significant competitive progress with Windows 8 against Linux on the server. Within the context of Linux it is especially important to call out **Google Android** which will likely be funded by Google for some time and represents an OS choice for mobile phones and phone architectures encroaching on the PC “from below.” Android can provide OEMs with the opportunities similar to Ubuntu, however Google is walking a careful line in providing Android where they can possibly lower costs for OEMs, or even subsidize the bill of materials for a device, counterbalanced by OEM wariness that Google will take too much control of additional revenue streams.

Beyond traditional OS competition we see a broad push to “out flank” or “disrupt” Windows. The motivation partners have to create new business models, differentiate their PCs and services, and to drive technology innovation in different directions creates a unique situation where competitors might look a lot less like an OS competitor and more like a subset of an OS or just way of selling things differently. As we plan Windows 8 there are three specific competitors that one might consider are taking an “innovator’s dilemma” approach, to cite an over-used expression.

- **Ecosystem value.** First and foremost, many are questioning the value of diversity of software and hardware relative to Windows. Many are claiming that the Apple model of a small set of great PCs with a well defined set of peripherals and all the software you need provided by Apple (and perhaps a couple of other companies for specific industries like design) are all you need. Some, such as **VMWare**, work to devalue the whole notion of hardware support by focusing on virtualization as the tool for management, deployment, and application development. This competitive approach is to essentially devalue the most valuable part of the Windows ecosystem. Our own transition efforts with Windows Vista inadvertently created the opportunity to prove or demonstrate this might be the case as many consumers and businesses experienced a world of Windows Vista without the value of the ecosystem built on 8 years of Windows XP because of poor device support. This also reinforced the opportunity for companies, such as **Cisco**, to add significant value to hardware and infrastructure, which then make for challenging integration scenarios for customers as well as a reduced importance to customers of the innovations in Windows. The key work we must embark on in Windows 8 is to build on the improved user experience, relationships and readiness in Windows 7’s development but doing so by delivering new and innovative opportunities to partners and

information to consumers in how the ecosystem is experienced in routine computing—Device Stage, Windows Update, Microsoft Update, Windows Live Partners, the Windows online community, and so on.

- **Browsers.** There are many that are developing hardware and devices that would claim “all you need is a browser” or sometimes “the browser *is* the operating system”. There is some truth, some irony, and some misinformation in this type of view. The reality is that nearly all consumers use a browser a lot and many cool and critically important PC experiences take place in a browser. The irony is that many of the folks saying that the only software is a browser describe a browser that includes a great deal of software beyond the browser—document or animation plugins, add-ins to complete the browsing experience, and even significant additional software not supported by standards or across different browsers (thus these are just more software beyond a browser). There’s an irony that browsing involves so much software—download managers, printing, saving files locally, and so on and each of these is an opportunity for us in Windows 8 to improve the connection between these “desktop” tasks and the browser. And of course the misinformation is the reality that we know from excellent product planning research that PC usage in the real world and how few PCs in reality are browser-only. As we plan Windows 8 we are going to be purposeful in thinking about how the system services we think of as Win32 become more readily available in the browser—we do not want to return to the approaches of the 90’s but think in new ways about the convergence of PC graphics and standards-based web graphics by bringing the incredible richness of technologies such as DirectX and to the web application developer. That is one example but through planning we want to expand this through many levels of the web development “stack” currently in the browser. By far, the company most representative of this trend is Google and in particular the business model that is supported by browsing (Search) that leads to substantial investments that continue to affirm the browser-only model (such as Gears), while at the same time using this business model to pursue alternate architectures, particularly related to mobile computing. In addition, we must consider that the browser does not stand alone and much like the ecosystem around PC hardware and software, there is an ecosystem around browsers—the ecosystem of sites, services, standards, and development tools. There is a significant challenge in how we approach this ecosystem, participate and also innovate. We’re seeing our competitors in browsing run into this as they follow a pattern of embracing standards while simultaneously extending them.
- **Phones and alternate architectures.** The iPhone is referred to by many as a new form of portable computer—“it has a real OS” many have said in reviews. The Google G1 phone running Android is likely to be made available in more PC or “handheld” form factors beyond the single-handed in-your-pocket screen. The desire for the ultimate device that has the power and capabilities of a PC along with the convenience of always on wireless connectivity is beyond alluring. It is what we all want as consumers. To deliver on such a vision many might say that Windows can never power such devices—perhaps that is a statement about “business model” or a statement about technology. Sometimes it is just competitors declaring “[Big] Windows will never be on a phone”. The reality is there was a time in the history of Windows phones (CE, PocketPC, etc.) where the synergy between “little” Windows and “big” Windows was technically

robust in reasonable ways. For a variety of reasons we diverged. Today we have hardware designers and phone company customers facing a choice between the OS that supports phone networks, voice, and other “phone” scenarios super well, but does not have the rich ecosystem support of Windows 7 and runs on a small set of hardware chassis, and Windows 7 running on a mainstream hardware platform with broad ecosystem support and openness. These platforms differ in bill of material costs, power consumption, and so on. Much like the “browser is all I need” statement there is a significant amount of extrapolation that both excludes Windows and assumes the Windows platform cannot compete. Our chip partners for Windows are working hard on bringing the x86 architecture “down” and we need to be there with Windows software. And at the same time we will strongly consider how to run Windows on alternate hardware platforms and learn what that would entail—we will work to bring “big” Windows to mobile chipsets, but we have significant groundwork to do before we know the practicality of such an investment. Our Windows 8 planning will most certainly take into account the role of sharing new code across Windows and Windows Phones, starting with the latest Internet Explorer as we are already working on.

As we can see there is a new wave of competition in addition to our traditional OS competitors. We could easily spread ourselves too thin as we work to compete on so many fronts. We should expect that as we plan Windows 8 we will be focused on where we can best compete with technology and where we can best compete with business approaches, and of course there is always a combination that must be worked on in unison. We can also assume that any of these competitive approaches will always be a journey and not over and done in one release. To compete effectively we will define solid deliverables in Windows 8 that deliver clear and compelling customer benefits that we can successfully point to as moving the needle in a visible way relative to the chosen competitors.

## Our Customers

There is nothing more unique than the Windows customer base and this is something each and every one of us should take pride in as well as feel the awesome responsibility that comes with serving nearly 1,000,000,000 people around the world with software in over 100 languages. Each of us individually represents a unique Windows customer whether we’re supporting our family PC, helping a local school use PCs, using internet kiosks on visits overseas, or the plethora of ways we all put PCs to work and play. In a matter of speaking it is almost impossible to categorize our customers or more importantly to try to label any single customer a certain way. Someone who uses a PC to drive an astronomical telescope is just as likely to use a PC to run their small business. The most enterprise savvy IT professional comes home to a digital camera and baby pictures. The farmer in rural Uganda using Skype to arrange a delivery of seeds takes time out to check football scores. This diversity of usage is at the core of the success of the PC platform—a way for customers to tailor their PC experience uniquely to different needs at different times in different places. Yet as we plan Windows 8 we still need to find a way to make sure we are clearly focused on meeting the needs of some of our key customer types. Who are some of the specific people we should consider? Who are the people that buy Windows from us?

- **Humans.** There is some humor in this, but we all too often are not as good as we need to be in considering that humans, or mere mortals when it comes to computers, represent the overwhelming majority of those who will interact with Windows 8. With Windows 7 we made strides in the right direction, particularly when it comes to the basics of launching programs, managing Windows, and dealing with exceptions in an otherwise smooth-running PC. And we also took new approaches to complex problems such as setting up home networks to share files and printers. We've made significant investments in new ways to interact with Multi-Touch and advanced technologies for speech and natural language. We lead the way in making sure everyone around the world, regardless of ability, can have universal access to a Windows PC. There is much to do and this goes way beyond things we can read about on blogs that would make Windows "dramatically easier"—we should fix the control panel layout, but we should not think that by doing so we have changed how easy it is for people to enjoy Windows and get work done. Yet even with our advances, Apple remains the benchmark for "ease of use" and in defining the bar for computer-human-interaction. Our aspiration for Windows 8 is to move this bar and for Windows to be viewed as the modern leader in the most basic elements of computing.
- **Enthusiasts.** *Seemingly* at the other end of the spectrum are PC enthusiasts. While they too are humans, when it comes to using Windows PCs, PC enthusiasts are in fact *super-humans* both in their knowledge of Windows and their passion for putting that knowledge to work to craft the ultimate PC experience. Many of us at Microsoft fall into this group and nearly all of us on the development team are enthusiasts. We do not fear (rather we embrace) regedit or command-line tools. We tweak and customize everything. We are quick to notice small changes each build. We say "our enthusiasts" but with our last release we showed how fleeting this relationship can be as enthusiasts went into hiding and were simply disappointed with our release. As a result enthusiasts look to alternate platforms to express themselves with computing—Linux, Phones, Macs, or just Windows XP. Very fortunately, with Windows 7 we have re-energized this customer group, but we're not there yet. In Windows 7, enthusiasts have embraced the product and are out there evangelizing it—we are starting to win back their support which is incredibly important in terms of the overall "influence model" of blogs, reviews, and journalism. One might even say we have Windows *fanboys* now. With Windows 8, what features can we deliver that will provide a meaningful and sustainable engagement with this most important type of customer? We should also consider that many enthusiasts are also IT para-professionals and so enabling the easy support of Windows for this group poses a unique and important challenge.
- **Enterprise IT Professionals.** Windows leads with consumers, but as a practical matter we rest on the supportive backbone of the IT professional. The IT Pro community is responsible for a significant portion of our revenue, and more importantly one of the key tools we have to moderate the impact of the PC run-rate on our business since IT Pros have the opportunity to establish long term relationships with Microsoft independent of the PC run-rate. The needs of IT Pros can often look like the needs of humans plus the needs of enthusiasts plus domain specific needs. IT Pros are themselves enthusiasts, yet they represent the needs (and manage

the PCs) of many humans. At the same time, IT Pros have the unique responsibility of managing and protecting the information assets of the world's enterprises. These needs drive a broad set of technologies which we provide uniquely in Windows. Security of course is both a core asset and continues to be, at least in the eyes of many, a potential liability of the Windows platform. Manageability is in a similar position and our significant investments in AD/group policy, management tools, and new technologies such as virtualization are industry leading but at the same time we often leave more questions than answers when it comes to deploying these in the enterprise. Stitching together our deployment tools into a solution still requires navigation through vast documentation. With Windows 7 we continued to enhance the features for IT Pros, again leading relative to competition. Our solutions for remote connectivity and branch office management continue to provide the most flexibility and options, but at a cost of significant complexity and arrays of options that make it difficult for us to be prescriptive—it simply won't work to continue to add more options, then more tools to be prescriptive and manage those options, and then tools to diagnose the failures due to conflicting options, and so on. With Windows 8, we must ask how we can work to alter the dialog with the IT Pro. How can we have a more "appliance-like" experience? How can we move the dialog to position Microsoft as a key provider of the dynamic, human-centric IT environment that we see becoming the goal of IT organizations. How can we change the discussion from the defensive position we currently have? As discussed earlier, we do not want to show off our organizational boundaries in the software we build. Enterprises look to the combination of Windows, Windows Server and the Office System as the solution from Microsoft. Every investment we make for enterprises has to be consistent and complete across our major offerings.

- **OEMs.** OEMs also wear several customer hats. OEMs employ procurement specialists who have the job of getting the best price from Microsoft for Windows as any procurement specialist would. OEMs are our technical partners who we work with to build great PC experiences. OEMs are ISVs who develop software to differentiate their hardware. And OEMs are our business partners with whom we together bring the Windows PC experience to customers around the world. Because of these many hats it can be quite a logistical challenge in working successfully. We should take a step back and do even more work in Windows 8 to deliver a platform that OEMs view as one they can bet their businesses on in terms of opportunities for differentiation—by implication developing opportunities for OEMs to make money from product and business strategies developed for Windows 8. One of the nice innovations in Windows 7 wasn't even a feature, but was the OEM Planning Forum which redefined how we work with OEMs when we think of them as our technical partners. How can we significantly improve the OEM relationship across these other dimensions? How can we improve on the first round of success of the OEM Planning Forum when we know expectations are higher now that folks have seen how such a forum can work.

The above represent some of the key customer "segments" we will focus on for Windows 8, but it is by no means exhaustive. As we plan Windows 8 we want to embrace this diversity without losing sight of solving specific problems for specific people—it is too easy in planning a release to just assume someone, somewhere, will benefit from a feature or to assume that if one of us *likes* a feature then



others will follow. How can we think of some important customers? How will we make sure that while designing for everyone, we also design for *someone*? Many will want to prioritize and ask how to make tradeoffs when customers have conflicting needs—we must avoid the tyranny of these false choices. If you have any doubt, recognize that we are here to serve the needs of the broadest set of people who benefit from PCs, even if sometimes they seem like the silent majority. Windows 8 will give us a chance to revisit basic assumptions about ease of use and how people interact with PCs, hardware, and software.

## Developers and Industry Partners

As a platform, Windows can only be as strong as the applications available and the partners who create the hardware and software that bring uniqueness and diversity to the Windows experience. This diversity is incredible and our ability to offer tools, APIs, services, and support to enable this diversity is of paramount importance. Despite this importance, over the past few years our message to partners and developers has not been as clear as we need. Our Windows APIs and developer tools have not been aligned very well and as a result our ability to reliably bring innovative functionality to the market has been hampered. And a reshaping of the center for innovation in PCs has taken place which we have not worked with effectively. Windows 8 is a new opportunity for us and by building on the Windows 7 base we can re-ignite these critical aspects of the ecosystem. Developers and industry partners represent the people that invest in the ecosystem and expect a return on that investment. As a key part of the ecosystem, we must work to define ways for these partners to implement their visions by building on Windows. As we think about partners we can think of them in the context of how we reach out to specific groups. There are developers who cross the entire ecosystem but share the common view of looking at Windows through the lens of the tools we deliver to them to build software. Hardware and software partners represent the companies (and enterprise organizations) that are bringing hardware and software to market, and each will see Windows 8 through the lens of the opportunity we create for them to provide unique solutions for their customers or opportunities for their businesses. And finally a new class of partner is the Original Design Manufacturer, ODM, responsible for much of the design and manufacturing of today's PCs and increasingly coming to resemble OEMs. The common theme across partners is our need to be deliberate in articulating how they can benefit—better service, differentiated products, and/or more profits.

**Developers.** It might be a bit tired, but we never tire of our CEO sharing his view of what's important as he *proclaims* "Developers, Developers, Developers---Microsoft Loves Developers!" Certainly the love is clear. What has been less clear has been our strategy for engaging developers. Our server strategy has not only been clear, but extremely effective, particularly at the enterprise level, with Visual Studio, ASP.NET, and the depth and breadth of support building on Windows Server. Our client strategy has been more challenged and that is where we will need to focus, and even revisit past decisions.

Our client development strategy faces the unique challenge of how Microsoft should "reconcile" or "support" or "coexist" with developing applications for browsers. During the development of Vista we made some strategic missteps in how we invested and innovated in browser-based development. As a result we found ourselves behind the evolution of competing browsers and behind on the implementation of what evolved to become customer expectations of "standards" support. With

Internet Explorer 8 we not only got “back in the game” in that dimension, but we also pushed innovation in new directions for end-users and developers. As discussed in the framing work for Internet Explorer 9, we must push hard on a number of dimensions for standards support, scripting, and in other areas for developers—all building on the IE 8 foundation. With this foundation we also have a significant opportunity to take a fresh look at how we would consider the assets of Windows 8 as part of the strategic opportunity we could provide for developers. For developers, however, our efforts must also include support for innovative development tools that support standards but also support the unique attributes for web development supported by Internet Explorer.

Building on the assets of Internet Explorer affords us the opportunity to bridge, and more important make seamless, the wide gap between the software experience for the web and the software experience for Windows programs. While we’re all familiar with the customer benefits of “web-based applications” what we’re seeing in the development of browsers is a fairly significant effort to essentially build larger and larger runtimes in the browser, which both validates the value of “client code” while at the same time removing the notion of “run anywhere” and replacing it with “run anywhere so long as it has the latest version of a specific vendor’s browser [and/or plugins]”. It is this latter point that opens up the opportunity to have a dialog about what we would do to redefine “browser-based” applications such that we show the value of using our Windows runtime to enhance this experience. We could think about how to do so by building on the extremely comfortable and familiar concepts of web development that exist today (HTML, CSS, Script) and the advances that are looking more and more exciting to developers (SVG, for example) and thinking about how those could build on core Windows strengths such as DirectX graphics, Windows User-Interface elements, and of course the very rich Windows input tools (ink, touch, speech). As with standards-based development, gaining alignment with our development tools is an opportunity that developers will expect and one which we will need to plan for in a collaborative manner.

There is a vast amount of software that is developed “natively” for Windows—not specifically in “native” code (v. managed code), but developed calling the Windows APIs. More than any other aspect of the post-Windows XP era, this is an area where our lack of strategic clarity has been challenging for developers. We have continued to produce remarkable innovation across what we think of as Win32 for clients. In just Windows 7 we have introduced Multi-Touch, Ribbon, Sensor platform, and a host of desktop integration features just to name a few. This is building on the vast array of native Windows APIs introduced in Vista at all levels of the system for networking, communications, storage, graphics, and more. Yet our strategy for tools has made it difficult for developers to tap into these innovations as we bifurcated the development message into “managed v. unmanaged”. The strategy we must embark on, beginning with Windows 8, will be one where developers can target the Windows APIs in our premier tools and languages without requiring them to use a specific language or specific library—and most importantly our tools will support the latest elements of Windows 8, and do so without penalties or constraints. For those that have been around for many releases of Windows, we might recall how a new release of Windows came with a new release of development tools and those tools showed how to support natively the latest advances in the OS API. We want to understand how to return to that type of rhythm and working closely with our partners in development tools, and for Windows 8 we will do this.

## Microsoft – Windows Team Only

While some may have said we have had a confusing message to developers, we have also failed to help Windows developers with their own engineering systems, market presence, and distribution. Over the course of the 1990's we create an incredibly powerful platform without much guidance to developers on how to write great software resulting in an array of security, complexity, reliability, and performance issues. These issues have been used effectively by competition of all types to position Windows PCs as fragile. As we contemplate the fusion of web standards programming with the benefits of Windows platform capabilities we have an opportunity to make client software as easy to own and manage as any web based application. We plan on expanding the notion of Windows API development to include development in web technologies as a peer, solving different problems in different ways. We view web technologies as a complementary subsystem that expands the richness of the overall platform. We won't solve all of these issues in one step, but we must be deliberate and definitive in what we will reliably deliver and visibly improve for Windows 8.

**Hardware and Software.** Developers cross the hardware and software boundary more than ever before. The variety of hardware devices and the richness of software provided with devices, and the range of software that operates better or requires unique peripherals is an incredible asset. When there is talk of using single purpose devices or devices that have restricted runtimes, one need only look at the amazing variety of ways our hardware and software partners connect to Windows, differentiate their products with Windows, and deliver innovation by assuming access to Windows functionality. Without hardware and software partners, there is little that is unique about Windows and we should think of these partners as our key differentiated asset relative to competitive operating systems. Our challenge is in engaging with partners, demonstrating the value we can provide to them, and in supporting them as they develop this value. We learned many lessons as we delivered Windows 7 and those were on display at conferences such as WinHEC and the PDC. Windows 8 is another opportunity to learn and improve.

Hardware partners in key peripherals of chips, motherboards, graphics, storage, audio, networking, etc. all have very high-bandwidth relationships with our team. At the same time there are places where we have struggled to have a consistent relationship where we place the whole Windows release in the context of these partnerships. This causes challenges, particularly with the partners that participate in more than one category or are working hard to develop single hardware components that provide a "system on a chip". One of the important efforts in planning Windows 8 will be to identify up front the specific areas where we wish to push hardware innovation and bring it to market with great support from Windows (across the whole OS experience). There is often a sense in our team that Windows can't drive hardware innovation release by release (in a single release), or that decisions on 2012 hardware were made years prior. In Windows 7 we proved that we become a better partner by committing only to things we knew we could deliver. We also saw the hardware industry make big shifts outside that long term planning timeline as mini-notebooks emerged seemingly out of nowhere. We must continue to learn how to navigate the challenges of engineering relationship management under different planning schedules and sometimes conflicting business goals. While hardware partners, just like everyone, would love to remove variables such as "what is Microsoft committed to in 2015" we must

## Microsoft – Windows Team Only

resist the temptation to communicate "information" with essentially infinite error rates—that does not make for a strong partnership since we know that *caveats* in our communication are never "heard".

A key tactic needed to improve hardware partner relationships is to better arm partners to do high quality support for new devices without the need for new “stacks” from Microsoft. Windows 7 helped us to learn a great deal about betting on new hardware with our work on multi-touch where a new technology that requires new software and enables new applications comes together in a new release of Windows. Another good example of creative innovation in working with partners is the role that Windows 7 Device Stage can play in bringing together partner efforts, making it easier for partners to differentiate and deliver value, and providing end-users unique integrated experiences while at the same time offering end-users an entrée to an amazingly diverse ecosystem. The experience we gained in the approach taken in Windows 7 can be magnified for Windows 8 and also improved upon in other aspects of the overall approach to hardware and software partnerships. Rather than push hardcore “requirements” that to partners often felt disconnected from their needs, we started with a focus on customer problems that both Windows and hardware partners faced. From that we developed a system that scales to the effort that partners could afford or were willing to invest in, while at the same time providing opportunities for differentiation. And by integrating this with a rich web service we build an experience that grows with the ecosystem.

The next step for us is to consider how we expand this type of effort, broadly defined, to software partners. Clearly there is massive interest in “app store” models. If we consider the efforts we can embark on with development tools and think about how we can develop canonical “phenotypes” of applications, we have an opportunity to develop rich tools that allow a community of applications that are easy to write, easy to distribute, and provide an opportunity for value creation among software developers. This will require significant coordination across groups, especially Entertainment and Devices, but is something we will want to consider as an initiative for Windows 8.

**ODMs.** For much of the history of PCs, the OEMs were very active in the design and manufacture of their PCs. During the development of Windows Vista our industry started to see a shift in the balance of where design and manufacturing takes place moving from the OEMs themselves to the new group of ODMs. The ODM provides a full and growing range of services to an OEM for the industrial design, manufacture, and even software load for machines. And we are now seeing ODMs branch out and begin to create their own brand names and sell PCs on their own, transitioning to OEMs. ASUS is by far the most visible with the expanding line of eeePC machines.

We have not yet established the right business and product development partnership with the ODMs, but with Windows 7 we are actively working with them for the first time in a significant way for all of Windows, and also doing so with their OEM customers. One might say that right now they are our partners because they are investing in the PC market and don't directly buy Windows from us, as the OEM is the Windows customer. This level of indirection has led to some challenges in making sure that new PCs have great software support and that we are influencing the choice of hardware components, as one example. The ODMs, looking to deliver a complete package to their OEM customers, end up looking to Linux as a way of providing a “full experience”. As we work to plan Windows 8, thinking

about how we work with ODMs as both business and technology partners will contribute to enriching the choice of PCs and hardware available for customers when we bring Windows 8 to market.

## Industry Trends

As we plan Windows 8 we are going to make sure we are deliberate in choosing which industry trends we will develop features and software in support of, and how we take these to market. No part of planning is more difficult and more open to angst—it always feels as though there are more trends than we can count and more risk in missing the next big thing. We can neither follow every trend nor can we assume that whatever we do defines the trends to be followed. If we are thoughtful and make decisions based on data and the potential impact of being right, then identifying trends effectively is one of the most important elements of the plan as we strengthen our efforts by focusing on these as thematic elements of the plan. A good example of this is Windows 7 and the focus on laptops—when we began the release we knew the data was showing a significant shift to laptops. This emphasized mobility, power usage, external displays, drive space, and memory patterns all different from desktops PCs. While no one predicted the rise of mini-notebooks as a massive accelerator of this trend, the bet on the trend put us in an excellent position. In addition, by operating with a well thought out plan we were able to take on new work and prioritize differently to adapt to the changing landscape—the agility we needed came from having a plan.

Looking at the landscape today, it is worth calling out a set of trends that our product planning team has been researching as well as trends we see from the business outreach for Windows 7. This is not an exhaustive list but provides the context for establishing the Windows 8 planning work.

- **Low-Cost hardware and explosion of form factors and price points.** Most decidedly the trend that has upended the industry in the past 12 months has been the rise of the mini-notebook. These machines running low-cost, low-power chips that are both highly portable and inexpensive, grew to be a phenomenon. Our business focused efforts have helped us to establish a Windows platform on these devices which we see consumers prefer. And our efforts at engineering Windows 7 for the performance characteristics of these machines appear to have been successful. Our challenge with Windows 8 will be the march “downward” of the hardware specs of these devices in an effort to save money, consume less power, and to get closer and closer to mobile phones. The most important aspects of this trend will likely be the possible changes to the Windows business model as described earlier. However the trend will also raise the questions of Windows on alternate processors as hardware makers work to provide differentiated “PCs”. The implications for Windows 8 include the role of x86 v. ARM processors and how we work with chip makers producing chipsets with integrated graphics providing a minimal level of functionality relative to our DX9 base hardware, or how we address the functionality provided by the BIOS or PCI bus if those are not available, as examples.
- **Cloud Computing.** Cloud computing is an opportunity for our work in Windows Live, hosted business software, and of course for Windows Server, as well as new services like Azure to come together. For many, however, the trend of cloud computing is a significant risk to the success of

Windows. Many believe that cloud computing implies lessening of the importance of a PC as all software is in a browser and all information is in the cloud. As we develop our plans for Windows 8, Internet Explorer 9, and Windows Live we will want to develop our unique way of talking about cloud computing in terms of the innovations in Windows that continue to show how important a PC will be, the way Windows 8 Servers and Azure will provide the backbone for cloud computing services, the role of incredibly cool browser technology and browsing, as well as the incredibly rich software and services in Windows Live. The combination represents the full range of software required to both implement the cloud and use the cloud. As we think about this trend we should not feel defensive about Windows, but recognize the reality that for many the idea of everything in the cloud is not a reality, but we have an opportunity to shape how the cloud can be perceived by showing the relationship across the entire software stack. This is true for customers and partners of all types.

- **Multi-core, Many-core.** Even the low-cost PCs are shipping with hyper-threading and we're sure to see multiple cores reach down into lower chip price points. Such is the inevitable march of technology. On the other hand, the mid range and high end chips continue to become dramatically more powerful and provide significant technology for us to take advantage of for new scenarios. The work on desktop concurrency and on support for 256 cores in Windows 7 is a great start. Our chip partners will continue to develop more and more cores and we should view this as a resource for us to use and empower Windows developers to exploit. Whether we exploit these assets for graphics, simultaneous voice/video/interaction, or parallel operations, we know that this trend is here to stay.
- **Mobility.** The mobility trend has exceeded our industry's best guesses and laptops have outpaced desktops, and even desktops have moved to all-in-one designs which are often laptop class hardware. The mobility trend in particular continues unabated with a dramatic focus to less expensive hardware and to 3G networking. The market has also made room for premium mobile hardware using fast SSDs, higher quality batteries, and ever-improving interaction technologies. For PCs we will continue to see consumers preferring laptops and businesses moving to laptops whenever possible, and even when a desktop is required it is likely to be built with mobile components. Mobility goes beyond the technology, scenario, and form-factor of a PC and also opens up the challenges of the sales model and channel strategy for Windows. In addition the "mobile" capabilities of a Windows PC vs. a mobile phone continue to be competitive speaking points showing us the need to maintain focus on improving power usage and speed to resume the PC with wireless network connection.
- **Proliferation of storage locations and devices.** As the world of media continues its move to digital, the desire for sharing remains strong. People and businesses will continue to share data using a variety of servers and services. Smarter devices store, retrieve and sync both directly and through the PC. This trend is both empowering and frustrating to customers, as the sense of control evaporates as quickly as their data when something "bad" happens. For Windows 8 we have the opportunity to work with Windows Live to address this issue in a more systematic way, protecting data by default while keeping devices in sync, and doing so in a way that is cost-effective for Microsoft.

- **Frictionless software.** The role of web browsers to obtain software for download has most decidedly become the norm for many people. Even the trip to buy tax software has been replaced by a trip to a web site and a credit card transaction. Games, utilities, and most everything but the most major packages are acquired by the web. Yet there is still a high degree of friction—the credit card, the Product Key, and then of course the challenge of getting a new or second PC and establishing the same software environment and configuration are all problems. There's a lot of friction in getting software onto a Windows PC. As an example, many downloadable programs are viewed by how many bytes to download, how many clicks to install, and how many UAC prompts are required. This is a measure of friction. This is all contrasted with the new bar for the acquisition of programs as set by the iPhone app store. As we look at Windows 8 we want to ask where there is friction in the system—friction to acquire new software, friction to buy a new PC, friction that prevents you from doing what you want to personalize your computing experience—and how can we reduce this friction. We've learned that we can use a service infrastructure and data that allows us to protect customers at the source of malware, before it gets on a PC, more effectively than relying on increasing friction which we know customers have become numb to. We've seen ourselves swing too far and introduce enormous friction yet still not solving the problem or just moving the problem around. The trend is clear in that innovation moves to customers where there is less friction. Historically we have looked at this through an architectural approach around defining new implementations for existing functionality, when perhaps a better approach might be to simply reduce functionality and focus on what developers might care about the most which is reducing friction and connecting to potential customers.
- **Consumerization of IT.** The days of people experiencing computing through the eyes of the pre-configured system at work are gone and have been replaced by people going outside the boundaries of their organizations (with or without permission) to seek the very best way to connect to customers, connect to information, or to connect to coworkers. Want to set up a quick mail box for your partners, then use a free web-based mail. Want to call a customer overseas while traveling, then use free VOIP. Want to have a quick place for you and coworkers to share files, then use web-based storage or collaboration. And of course as employees are doing more work from home on flexible schedules the dynamics of the corporate network are changing dramatically. Enterprise IT managers are struggling to keep up with the demand for services like these and more important struggling with the security implications as employees use more and more services outside the firewall. At the same time cost pressures are dramatically changing the service level offered by enterprise IT. Many companies are moving to self-support models or are even considering how to get out of the model of supporting PCs directly (such as how phones work in many businesses). This trend is going to continue and has significant implications in the types of investments we make in enterprise management, security, and the notion of dual-role machines that serve both home and work, or better said inside and outside the firewall.
- **Virtualization.** Virtualization is clearly a trend that has reached more than critical mass in the enterprise data center and Windows Server 2008 R2 provides an amazing level of support.

We're also starting to see the strong desire to bring together all Microsoft's disparate technologies that provide for "virtual access" to applications—remote desktop, desktop virtualization, and application virtualization. These are very different technologies that at the highest level provide ways for desktops to seamlessly access software without impacting the local PC. The trend to find ways to "virtualize" the enterprise desktop will continue because of the savings in direct costs for support and maintenance. Windows 8 will likely see a strong desire to deploy enterprise desktops as entirely virtual images—where updating or replacing a PC is a painless operation based on managing a virtual image from a data center. Yet we are not clear in which of our technologies will yield the best experience and how customers can navigate our offerings. We will also continue to see pressure from virtualization in consumer PCs, often utilized as a suboptimal solution to known problems such as instant on, application compatibility and isolation. For us, virtualization, in some form, will likely hold the key to how we can substantially change the model for applications while maintaining compatibility in some form (perhaps a lesson learned from how Apple made the transition to OS X).

### Defining Success

The previous sections developed a context building on Windows 7 and then describing the business, customer, partner, and technology environments as we plan Windows 8. As we plan together, it is important that we have a shared view of what success looks like as we build Windows and bring it to market—we're all aiming for one step above winning reviews, gaining share, or growing revenue. As a product that is both a platform for partners and also a consumer product, defining this success is complex. We must be careful not to subject ourselves to the *tyranny of or* as we must deliver on both the platform and the experience, on quality and predictability, on the business and the product, and so on. As a business that serves consumers via PC makers, and IT pros via enterprise sales, we must also effectively serve both and not continuously ask which one is the priority. We must also not try to distill down a list of success factors into "the most important ones" or "the ones that we *really* need to pay attention to". We must not just assume we can turn situations into absolutes or "more is better". Rather, we must recognize and embrace the complexity and subtlety that comes from working on a product with the potential to benefit so many.

To establish long-term and sustainable success, as a team we establish the following success criteria:

1. The complete Windows experience, including the Windows operating system and Windows Live, represents the best user experience available on a PC and device—whether the person using the PC is a novice or expert, at home or in the enterprise, an "end-user" or IT professional, people who use Windows, love to use Windows and find using Windows a pleasure. Key ingredients of this experience include knowing that you can trust your PC to be safe, secure, and private.
2. As a platform, Windows is only as good as the applications that run on it and the experience of running applications. We build a platform, but the most perfect APIs and system services in the world are not successful unless they are used by developers to build unique and innovative applications that are only available on Windows. The ownership cycle of acquiring, running, maintaining, re-acquiring, and disposing of applications must be the very best environment of



any competing platform. Communicating this experience to people around the world and evangelizing the opportunities to developers are key aspects of being successful.

3. As a platform, Windows provides the best platform for robust integration of hardware. The broader the selection of hardware (with supporting software) available on Windows the stronger our platform becomes for customers. Our ability to uniquely support devices, supporting both customers and hardware partners is an asset of Windows so long as we do so more effectively and with higher quality than any competing platform. Through our outreach and evangelism we demonstrate to customers the value of peripherals and choice and provide a platform for IHVs to innovate.
4. As a platform, we will offer developers with the most unique and useful combination of APIs and system services, always maintaining compatibility out of respect for the investment developers have made in Windows. The way we express the Windows' assets to developers is through APIs. Managing the evolution of these APIs and maintaining compatibility release to release are key tenets of how we engineer Windows.
5. As a platform, Windows provides a broad set of APIs that are expressed in ways that can be used by tools programming in any language. Our development tools must be designed to consume these APIs in a language-specific manner, but not in a manner that precludes the full breadth and depth of functionality. Today we have a broad schism in how developers can view the Windows API asset. Some developers see things through the lens of C/C++, while others see this through COM, and others see this through class libraries that present new abstractions and a filter through which Windows can be programmed. While Windows has many (too many) API styles and conventions because of history, the way we will manage this is by respecting the legacy of Windows and being clear with developers about the APIs in Windows that represent our "best foot forward". We expect the Windows APIs, across all layers of the Windows system, to be considered our asset and our tools to provide access to the breadth of APIs.
6. Windows is the most secure, reliable, scalable, accessible, global, deployable and manageable client OS and the best foundation for the full capabilities of Windows Server. Our product stands for these core tenets that are required for all customers. Our enterprise customers will evaluate Windows extensively with regard to these product "abilities" and we will continue to up the ante each and every release. The ability for customers to validate the success of the platform against their definition of success in these abilities, and to do so in an industry leading cost structure is how we evaluate success.
7. Windows Live delivers compelling software + services for consumers and developers, and does so by building on Windows with Internet Explorer. While we have many services in Windows that are used by nearly every PC in the world, we are also extending the Windows experience with Windows Live. The software + services in Windows Live are a significant part of the *complete Windows Experience* whether we are having a dialog with developers, end-users, or business customers. In addition Windows offers services such as the Windows web site, Device Stage metadata, Windows Update, WinQual, and others. These services for partners are part of our platform and we will plan and execute on them using the same methods we use for the Windows client so we can present an integrated value proposition to our customers and partners.

8. Windows offers OEMs, IHVs, and Developers the best opportunities to robustly deliver differentiated, valuable, and profitable experiences to customers. While we develop a broad product with a specific brand definition, we must also increasingly provide clear and low-cost means for partners to develop differentiated products and services.
9. Windows offers enterprises complete technology solutions that are flexible, powerful, yet cost effective at meeting needs. Our infrastructure offers high degree of consolidation opportunities for enterprise data centers, while delivering an excellent experience for a more distributed, mobile workforce.
10. As a business, we innovate in the ways customers can learn about and acquire Windows, while executing superbly at the OEM business model of today. The nature of selling Windows changes as much and as fast as the technology of Windows and we must strive to be just as innovative as possible.

## **Evolving the PC**

Our first step in structuring Windows 8 will be to develop an understanding of how we think of the PC itself. Nothing will undergo more change in the short term and long term than our traditional definition of the PC. The clear separation we see between server, desktop, laptop, and mobile phone will be replaced by a continuum of computing devices. If we stick to “Windows is for PCs” and we define PC through a traditional lens, not only will we miss out on opportunities to increase the places where Windows can be valuable to customers, but we will also run the risk of having Windows defined to be applicable only to yesterday’s devices. At the same time, taking our focus to the extreme in any one dimension—that is focused too much on server/cloud or too much on mobile—will leave some computing opportunities “on the floor”. We will develop our Windows 8 plan to deliberately tap into the capabilities of micro-processors across a broad spectrum of devices and either provide directly, or provide opportunities for, the development of Windows capable experiences on a variety of “form factors”.

**Mobile Devices.** At the one end of this spectrum is the role of Windows for mobile devices. Today the convergence we are seeing between devices running x86 Windows and devices running ARM/CE Windows is happening at an incredible rate. There is a strong demand for Windows devices to assume the positives associated with CE-based Windows (particularly battery life and cost) while maintaining the positives associated with Windows (devices support, development tools, ecosystem, etc.) The engineering challenges here are, of course, significant. The approach we are going to take involves creating a strong partnership with the Windows Mobile team, first and foremost. As part of investigating Windows 8 possibilities we will investigate the possibilities of support on alternate hardware platforms. Some are going to be skeptical of Windows “reaching down” to low-power, low-price chipsets just as some are skeptical of new chipsets gaining traction on the low end. On the other hand, few would argue that the pace of increasing computational power in what we consider a phone today is rapidly approaching today’s mini-notebooks and we already know Windows runs effectively there.

**Converged OS.** As we see the phone and PC start to take on the characteristics of a single OS/single system, we will take an approach similar to what competitors have done which is to look at important subsystems and decide where we will have the most to gain in code reuse. We will start by making sure that Internet Explorer 9's rendering (HTML, CSS, Script, etc.) forms the best foundation for mobile browsing (in a partnership with Windows Mobile of course)—this continues the work already undertaken with previous releases of Internet Explorer. With this effort we will also solidify the approach we take on graphics such that there is much more sharing and synergy across the phone and the PC. We believe that we can bring the Base Windows OS in combination with the display subsystem and the capabilities of the rendering engine of IE to a performance level that forms the foundation of a modern device, certainly on x86 hardware and potentially on other chipsets. The transition from today's CE-based OS to a Windows-derived OS is a major undertaking. Windows is committed to developing these core components in partnership with the Windows Mobile team who will ultimately define the approach and architecture to building our phone devices. This is a "work down" process from the essential end user and partner visible interface (IE) based on what we learn to be feasible as we continue to learn and leverage our work following x86 devices "down" from PCs.

**Managed and Unmanaged Clouds.** At the other end of this spectrum, server computing is undergoing this same type of "collision" of computing infrastructure. Today we generally think of a clear separation of the data center for cloud computing and the data center for enterprise computing. It goes without saying that we believe we need to have the scale of cloud data centers available to our enterprise customers and that our investment in tools for management and operations should be applicable to cloud computing. Virtualization technology, both virtualization and managing large numbers of virtual machines in a data center, is a key technology that will define the connection across this spectrum of computing and enable us to enhance the Windows value proposition rather than cede it to competitive platforms. The Server of course plays a role in managing and deploying virtualized clients in addition to running server applications.

**Mainstream PCs.** In between would be the space we normally think of as the "PC" in various form factors such as game machines, desktop, all-in-ones, and laptops of various sizes. As we've seen the mini-notebook as essentially blurred this distinction and caused some consternation among industry insiders as to how to evaluate these machines technically. Regardless of how to categorize them, we know one thing for sure—Moore's Law applies to this hardware as well. We've already seen the next round of CPUs get faster and we've seen discrete graphics appear and be touted on these machines with ever-increasing memory and storage. We should expect the low end to continue to reach down, thus continuing to rely on our engineering effort to maintain the system requirements at a constant level, which by definition will mean performance improves release over release.

**High End Experiences.** Another great example of an untapped potential is the "full room" PC. That is a PC that uses the capabilities of Windows to take advantage of multiple displays, rich input/output devices, and a wide range of connectivity to deliver a computing experience that can encompass the needs of an entire room. Through such a PC a meeting room, a classroom, or an office can be transformed into a collaborative workspace. Such a PC would have cameras, touch screens, audio and video all coming together in a unified experience. Alternative scenarios for high-end experiences might

## Microsoft – Windows Team Only

include a “Theater PC”, professional workstations, or extreme gaming PCs. All of these will be used by customers and our challenge is to investigate features and support for any specific end-to-end scenarios.

**Shared Computing.** Markets that are more price-sensitive look to squeeze the most out of every piece of hardware. This could be a PC shared by members of a household, thin clients connected to servers or “cloud PCs”, to booting from a personal removable (USB) drive, to multiple monitors shared by multiple users connected to a single CPU, to split screens and multiple mice. These are some of the use cases being considered and/or deployed in emerging markets and education. Together with the Unlimited Potential Group, we must have compelling solutions built on Windows for those markets.

Across the spectrum of PC usage today are opportunities for us to radically improve the applicability of Windows to strictly-defined computing scenarios—so called *appliance* scenarios. The way Windows is installed, updated, and maintained are all assets in many scenarios and liabilities in many places where innovative companies consider alternate platforms to be more applicable. Today we think of these devices as set top boxes for television, task-specific workstations in settings such as hospitals, control or customer service machines in factories or call centers, and a broad array of consumer electronics devices for home entertainment. Each of these has opportunity for us and in most cases we have not been prescriptive enough or listened to customers enough to build a great experience on the very technology we have. A great example of this is the set top box for TV. We have all the technology in Windows to provide an experience much better, more customizable, and easier to develop for as evidenced by Windows Media Center. Yet time and time again we see our complete offering and business model come up short. The basics for support of lock down, reset, and control prevent Windows from appealing to customers. For example, most every consumer device includes an ability to restore to “factory defaults”, often at two levels (preserving settings or not). Windows does not have this complete capability automatically built-in today and we rely on OEMs for ad hoc solutions to this problem that do not add up to a value proposition. We can do much better than we are doing, and as we learn about each of these areas we can see that anything we do applies equally well to a traditional laptop or to a potential new business model.

In many ways our opportunity for how Windows can be more applicable in more places is to consider this table of Windows system requirements over time. To many customers this is a core weakness of Windows—the seemingly unbounded growth of Windows over many releases. With Windows 7 we engineered an *unprecedented* product in that it is the *first release of Windows that did not effectively double the system requirements of its predecessor*, and in fact we even reduced the requirements for disk footprint. As we look at the historic system requirements below, let’s consider it a challenge to do even better for Windows 8.

Approximate/Recommended System Requirements						
H/W	Windows 3.1 (1992)	Windows 95 (1995)	Windows 98 (1998)	Windows 2000 (2000)	Windows XP (2001)	Windows Vista and Windows 7 (2006 and 2009)
<b>CPU</b>	386	486	66 MHz	133 MHz	300 MHz	1 GHz
<b>RAM</b>	3MB	8MB	16MB	64MB	128MB	1GB
<b>~HDD</b>	14.5MB	55MB	200MB	2GB	1.5GB	40GB (16GB)
<b>Graphics</b>	VGA	VGA	VGA	VGA	SVGA	DX 9

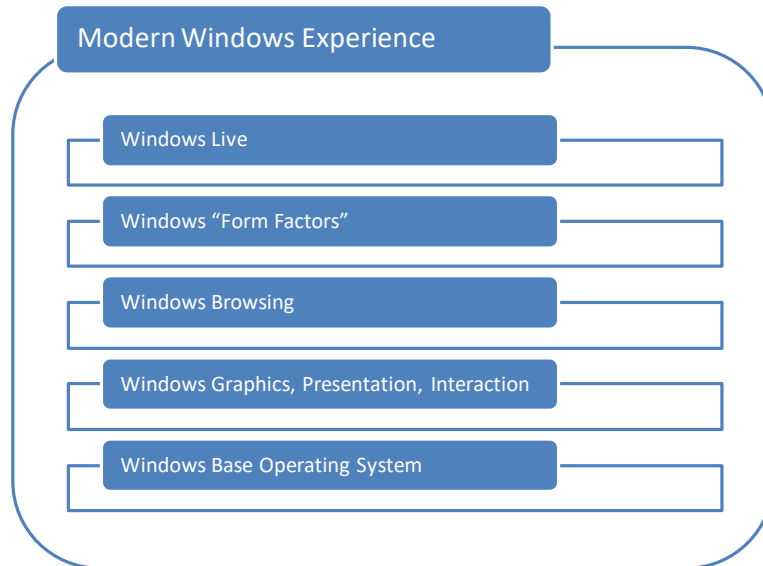
The PC is a marvelous device. The changes going on at every level of the hardware stack are unprecedented. Windows 8 must create a more flexible foundation while at the same time we must be more deliberate in defining scenarios that we have engineered for and support.

## Structuring Windows 8

Through this memo we’ve established a baseline of context with which we can all move forward and begin to develop a set of scenarios and features that will position us to best achieve success. As we developed the Vision for Windows 7 we created over 40 different core scenarios that would come to define the release. These scenarios followed from the planning work and the vision pillars defined the logical organization of these scenarios. Picking the starting point for planning is always a bit tricky. We don’t want to start with technologies and then hope we can define scenarios that use the technologies. We don’t want to be so focused on scenarios that we end up with redundancies or inefficiencies in how we build out the underlying technologies. This memo kicks off our planning by prescribing a conceptual view of Windows 8. From this we will derive themes for planning and execution.

As this structure is considered, it is important to keep in mind that we will not think about building Windows 8 following this structure nor will we organize the team this way. It is a way for us to think conceptually about the technologies and to group scenarios into a logical “layering”. This is an architecture for how we will think about planning. This conceptual view also provides the structure for how we will work with other groups in the company—where do they “connect” to Windows and how other products and groups support the Windows 8 effort so we can achieve the success described above. This conceptual view necessarily incorporates the PC operating system, the server OS, Windows Live, and of course Internet Explorer. Within the PC operating system we include all of the code and all of the groups that contribute directly to shipping Windows. The server operating system defines workloads and roles that are included in this conceptual view. And Windows Live is defined to include the consumer experience and the platform capabilities.

Windows 8 is conceptually defined by 5 entities which collectively represent the **Modern Windows Experience** as shown below (not to any scale). Each of these components is related and moving from the bottom of the diagram you can see how each subsequent element builds on the capabilities of the previous. *Scenarios*, as we will define them for the Vision, will by design be vertical slices through this diagram and will be developed by multiple teams and will be developed under the umbrella of a *modern* experience. As we describe these in detail the descriptions will use example technologies, features, and scenarios but no attempt has been made to be exhaustive.



The **Modern Windows Experience** is how we design, develop, and communicate the value of Windows. As technologists we often focus on the technology components within the whole experience, but we should not lose sight of what the vast majority of people see when they look at Windows – the user experience, end-to-end, top-to-bottom, from the concepts to the details. We cannot succeed without being leaders in how we define user experience, how we express this as the user interface of Windows, and customers seamlessly move from one scenario or task to another. More than any technology aspect of Windows, delivering on an innovating, pleasing, and productive experience is something we must excel at implementing for Windows 8, and we must do so in a way that surpasses the competition. As we embark on defining Windows 8, we should not consider success distinct from leading the way in the creation of innovative experiences that revolutionize computing. In fact, as we communicate the value of Windows to the marketplace through the moniker “[Life Without Walls](#)”, which we consider our *brand promise*, we will do so choosing to highlight features that best illustrate this complete Windows experience—a *modern* computing experience in all forms. This will come from all of us thinking about “One Windows” and making tradeoffs in experience that optimize for a greater, more modern, whole and not necessarily developing what appear to be locally optimal, domain-specific, solutions.

This experience will encompass all Windows-based computing across all screens, form-factors, clients or cloud. It will be important for us to consider how the design, branding, and overall experience are expressed in a coherent way. We cannot be espousing a life without walls while we are building barriers to usability and comprehension within our software.

As we look to achieve excellence across the technology “stacks” and the experience that envelopes them, we must resist the temptation to “change” without a clear benefit. Change that is only 10% better is often just 100% different. We’ve seen this time and time again when customers call for “classic mode” or when we seemingly “remove features” from one release to the next without reason. Change is necessary to improve the product and necessary to deliver on new scenarios, but it must be principled, well-articulated, and provide clear benefits. At the same time, a key part of planning will be

## Microsoft – Windows Team Only

to realize that improving the legacy experience might ultimately be more beneficial than introducing a new experience, and certainly we must be careful in how we deprecate features (APIs or user interface). The measure of success in moving forward is that change is embraced and eagerly adopted by customers, not resisted or postponed by “compatibility modes”.

The **Windows Base Operating System** forms the foundation of the hardware/software boundary and the foundation for functionality that defines the base computational elements of the system: memory, storage, processes, networking, drivers, hardware connectivity, identity, management, authentication, and so on. The base OS also forms the foundation of our Windows Server computation load and thus is the part of the OS that must be engineered for scale up on our largest hardware as well as scale down to our smallest and most minimal computing platforms. Some scenarios, such as energy management, have become surprisingly important at both ends of this spectrum. The challenge to optimize for both of these scenarios from one code base is significant and constitutes a continued undertaking with Windows 8.

One element of our approach for moving the base OS forward is to stay on top of the key technologies that impact the system that supports all the higher level aspects of Windows. Examples include CPU and chipset, power management, multi/many core, virtualization in its many forms, USB 3.0, new storage technologies for servers and devices and their accompanying file systems, encryption key management, Bluetooth, and more. Our approach to these technology-based innovations is to work closely with our partners and help them to make measured bets on the technologies we believe will definitely come true. Another aspect of our work is to better enable our partners to bring new technology to market without relying on an entirely new “Microsoft stack” each release. This dynamic creates too many false trade-offs for us in terms of one-off projects and disruptions when we would rather be focused on value creation in major versions of Windows. We need to build our skill in helping partners do a good job on the early technology delivery so that standards can be delivered in our Windows releases. In our approach to technological innovation, we want to take care to not deliver technology for technology’s sake and instead focus our energy on technologies that deliver important new end to end scenarios.

The base OS is will also undertake how Windows manages following x86 into non-PC devices and the possible support of non-x86 platforms—what does it take beyond recompilation and how do we make progress. Thinking this through as a scenario instead of technology is important as we think through the Windows wide ramifications of delivering this. In addition the scenario helps guide key decisions around booting from different storage types, if we are to have any chance of keeping Windows applicable to the broad array of devices and scenarios required of the Windows business.

In many releases of Windows we have taken for granted that the base OS moves forward based on technology initiatives, such as the implicit need to provide support for specific networking technologies, drive for battery life, rich media support, and delivery of a standards and browser based development environment. By thinking this through from the standpoint of delivering “the browser down” we can guide key trade-offs that allow us to best stage a long term engineering plan.

## Microsoft – Windows Team Only

An example scenario pertaining to the boundary of software + service is the work we can do to improve storage and file systems as a foundation for desktop search. Today these are two parts of the system that most every customer experiences through one lens, yet our connection only exists well “above the file system”. We’re all cautious about undertaking an endless project on broadly unified storage. But we also have so much we can do to pragmatically tap into the infrastructure and user experience we have created in Windows 7 and bring a much more performant, reliable, and integrated approach that spans across computing scenarios.

Another interesting example is the work we have all been going through, which our customers go through for the life of their PCs, is trouble shooting. Watching the *win7shp* alias is an interesting experience as a multitude of different logs, log generating tools, and approaches are described for diagnosing the problem. Some of these are designed only for admins while others are commonly suggested to end-users for trouble shooting. Troubleshooting is a scenario that we should think of in more of a unified fashion and consider it as part of this work—of course we never want there to be problems in the first place, but great engineering also presumes some problems will happen. One of the key beliefs we should have in terms of improving Windows overall is that we should develop technologies from the perspective of the whole system and not specific subsystems because ultimately the whole system is what is experienced by customers. We should also solve the problem, not create a framework that assumes everyone else will revisit and “snap to” a new model.

Also illuminating is the acquisition and management of Windows itself and Windows applications. New technologies are making it feasible for Windows to be run in new ways (boot from VHD is an example) yet we continue to think primarily as Windows being delivered through lengthy setup or sysprep. Applications are delivered with similar assumptions from 1995. Over time an endless amount of complexity and compromise have left Windows uncompetitive compared to other application model technologies like HTML and Flash. We’ve added complications like product keys and anti-piracy. We need to rise above our technology focused thinking and deliver a greatly simplified experience.

Our work on system security helps to illustrate many challenges in our approach. If we consider the scenario of helping end users cope with a world of malware, we have had a technologically driven approach focused on preventing compromise of “the system” and between “user accounts.” Meanwhile regular people are getting phished and much malware installs without intervention by the system as shining examples of applications that run perfectly well as standard user. We need to start thinking like an end user and deliver an experience that is simple and provides protection from the bad guys. Thinking a little more broadly we would include all aspects of PC safety and look at this through the lens of Windows Browsing where the majority of these attacks are entering the system today.

The major scenarios outlined here are examples to help describe how we want to shift thinking from a technology focused approach to an integrated approach where we know the technology in depth, know user needs in depth, and use our creativity to fuse modern and simple solutions to problems that delight the customer. These are examples and do not represent specific goals or starting points.



The **Windows Graphics, Presentation, and Interaction** forms the foundation for rendering rich information on the screen and interacting with people using a broad range of technologies including: DirectX for 2D/3D/Text/Animation/Gaming, high fidelity audio and video playback, ink, touch, cameras, sensors, printing, as well as the array of controls and user interface components. While we have many investments and a breadth of functionality, we have not done all we can do to bring a coherency to our overall strategy and this has weakened our strategy for developers which has in turn caused even more confusion in how we approached the implementation.

One of the biggest bets made in Vista, the benefit of which has been realized in Windows 7, was the bet on a new graphics driver model and with that moving the state of the art of Windows graphics from GDI to DirectX. With Windows 8 we are going to build on this even further and assume that DirectX forms the foundation of the work we will do in low-level presentation. A side-effect of this will also allow our browser and development tools to assume that DirectX is our common rendering surface and will no longer need to duplicate this functionality or repackage it in libraries to be used for developers. The graphical model for Windows will be based on these technologies, while of course supporting and bringing forward our existing code and customers.

As we improve this area for Windows 8 we will also continue to support the developer community that wishes to build more “self-contained” applications as evidenced by the wide support and interest in the Silverlight runtime which provides graphics and media capabilities coupled with high-quality development tools. As we plan Windows 8 we will assume that Silverlight will be a very interesting runtime for developers and will establish a class of OS-independent and browser-independent applications. We will support Silverlight by providing DirectX on Windows Vista, Windows 7, and Windows 8 and we believe that Silverlight benefits by using the existing OS implementation when it is running on Windows PCs.

The ribbon was a significant investment in Windows 7 and we expect to see this user-interface metaphor extend to more of the Windows experience, and Windows Live in particular. Evolving the ribbon to enable the much richer rendering we will assume is available is something that will highlight the advances in the platform overall. We should not underestimate the business value that comes from a consistent experience in applications across the ecosystem that can be supported with easy-to-use and visible technologies such as the ribbon.

Microsoft has led the industry in investing early and boldly in speech, ink, touch, and a wide variety of input methods and recognition technologies. Windows 7 brings these to a new level of completeness and of global scale. At the same time there is much we can do to bring these input methods to life across the entire user experience of Windows and to maximize their value on different form factors.

We will also require tools that work to target these APIs and this “layer”. We’ve engineered 11 releases of DirectX graphics but have never delivered dedicated tools for developers to take advantage of these graphics, and more importantly our flagship tools do not provide developers any support to integrate this technology natively in their applications. Our advances in this area, while often opportunistically supported in libraries, are not always well-integrated or supported with first-class tools integration.

Libraries that “re-model” the APIs and provide a different conceptual framework can do just as much to hinder developer productivity as to help, as samples, documentation, books, training, and so on need to be done for multiple conceptual models, not just multiple languages. And even less appealing, our tools often choose to duplicate the code in Windows within the libraries, thus imposing a performance penalty for using the latest Windows capabilities.

The **Windows Browsing** area represents the full browsing experience and platform. Conceptually, if we combine the Base Windows OS and the Windows Graphics, Presentation, and Interaction areas with Internet Explorer we define essentially a browse-only PC. As we build Windows 8 we are going to be very focused in recognizing the preeminence of Browsing for consumers, IT professionals, OEMs, and most importantly developers. We have very ambitious goals for Internet Explorer, building on the foundation of Internet Explorer 8. We will be measured in how we approach the work, but no one should be confused that we are going to view the success of Windows application development through the lens of Windows Browsing. Planning for Internet Explorer 9 is already underway and is covered here only briefly.

We will begin by building the best browser for real people who browse the web. Internet Explorer will deliver the most responsive web experience—focusing on not just raw performance as measured by micro-benchmarks, but performance as experienced by real people doing real web browsing. Internet Explorer 9 will also continue to be the browser that is the most trusted, safest, reliable, and respectful of the end-user.

As we develop Internet Explorer we will do so making a strategic bet on the underlying Graphics architecture of Windows, transitioning IE from GDI to DirectX. In the course of doing this we have the opportunity to significantly improve the quality of rendered pages. A key motivator for this investment will be to provide the richest implementation of early-stage standards such as SVG. We will implement standards and compete based on standards based browsing. However, we believe that developers will want more and more from the web as illustrated by what competitors are adding to browsers. The opportunity for us is to look at the assets provided by Windows, not just in graphics but across the entire platform, and determine how to best surface these in a safe and secure manner.

Ultimately, our goal is to enable a seamless development experience such that applications developed using what we think of as browser technologies today (HTML, CSS, SVG, Script) appear, function, and interact with what we have traditionally viewed as “native” Windows applications. It is not far-fetched for us to think that someone might build a custom media player, custom explorer, or social networking “client” in the APIs surfaced as Windows Browsing.

By investing here we are not saying that Win32 is a “legacy” API, but we are recognizing that the vast majority of software development taking place today is done through the lens of browser-based technologies. The competitors making browsers are not sticking to pure web standards but are extending the standards by providing “rich client” code and APIs as part of their browser distribution. We’re going to do the same because it is the best directions for our customers. Our approach is to think

## Microsoft – Windows Team Only

of the browser as a way of expressing the richness of the PC platform. We will of course do this by architecting the right solution for the safety and security that people have come to expect.

As we think about the term “app model” we will think about applications written to this set of APIs and capabilities. The many elements of the model such as deployment and management must be thought of through the lens of web applications, where most of these concepts simply do not exist. The impact on system performance over time is minimized or pushed down to near zero, encouraging people to try new applications experiences without risk or downside. At the same time we have an opportunity with Windows Live to provide developers capabilities such as synchronization and roaming, which we can look to make available seamlessly across the browser environment and Windows desktop environment.

Other platform technologies also need to evolve as we refine our definition of application. Web delivery of rich application experiences challenges our views of security for software running in the system. Managing online identities and integrating them with applications and services create tremendous user value to those users running out of ideas for “strong” passwords. This is all enabled by great networking and cache support across a wide variety of bandwidths, offered at varying costs, from multiple providers, in diverse “roaming” situations. Then there is how it will all work when the user experiences varying connectivity. We often think of the binary offline or online, when in reality we know from years of developing enterprise email, that even with perfect connectivity the best architecture is one that is “always offline” yet appears online whenever the connectivity is there.

Because of our commitment to have a release of our development tools at the same time as Windows 8, we will work hard to develop an approach that demonstrate tools support for developers to take advantage of these capabilities. We will also continue to have tools that are “built-in” to the IE experience which will enable us to have a spectrum of tools support for many types of developers.

Thinking collectively about the Base Operating System, Windows Graphics, and Windows Browsing we can see the opportunity to partner with the Windows Mobile group to develop a future generation of mobile computing. As we continue down the path of planning Windows 8 and the Windows Mobile team solidifies future plans, it will become clearer where we will see code sharing and reuse. We know for certain that the bet with Windows Mobile starts with Internet Explorer 9 and we’re assuming that as we execute on Windows 8.

The **Windows “Form Factor” Experience** represents the way that customers primarily view Windows client computing devices--primarily PCs of many industrial designs today. We use this phrase to encompass the overall user-interface and tailored experiences that are required for end-to-end scenarios, hardware form factors, and the overall interaction model. As we plan Windows 8 we are thinking of this in terms of “form factors” because so much of the overall experience will be determined by the type of hardware or the specific scenario in which Windows and hardware are being used together. This conceptual layer has many important technologies: explorer, program launching, window management, personalization, gaming experience, home networking, mobility, television, device management, enterprise management, multiple-monitors, communications, and so on. We can think of longer term form factors such as television settop boxes or Windows Mobile, which require

collaboration across the company to “use” Windows as the basis for those form factors. One way that we can think of form factors is to think of the “three screens” – the computer screen(s), the phone screen, and the television screen(s). To complete the full picture, we might also say Windows Server also represents a form factor, which is expressed as server roles and workloads.

Planning Windows 8 will require us to develop the scenarios that clearly identify the role of various form factors. Obviously the most important form factor will be the laptop and how we build on the Windows 7 improvements in this area. There is so much we can do for the person using a laptop to launch programs, manage windows, play games, incorporate media, and of course browse the web all using the vast array of software, peripherals, and personalization available today.

But we should not just think of form factor as a type of hardware. Form factors also represent key scenarios for Windows 8 as well. Many of us think of a PC at home dedicated to media playback, streaming, or storage, or a combination of those. The role of digital media in all forms with the PC offering an amazing experience to consumer digital media, but also to organize, store, and share, represents an experience that we must deliver in an innovative manner. The Gaming PC is another example of a form factor. This is one where the competition is perhaps best illustrated at the extremes of the living room console and the dedicated portable device. Yet we know that many of the innovations in performance and quality that we all benefit from have their roots in meeting the needs of “gamers”. This scenario also happens to be an historic strength of Windows which we should be clear about how we energize and improve the platform and experience for these developers and end-users.

One aspect of being scenario-oriented in how we think of this conceptual layer of Windows 8 is the role of the Explorer. We have tried to use the Explorer to be all things to all people. In doing so we’ve pushed the metaphor very hard and have provided the one of the most significant Windows “apps” around. For many people the explorer provides a great set of functionality, but we know we have much we can do for this functionality to be more broadly accessible. At the other end of the spectrum, our PC enthusiasts feel Windows still lacks, or somehow removed, a great deal of “power user” functionality. From a form factor perspective, enthusiasts are running multiple large monitors, connecting to network shares for massive amounts of storage, and using an array of local storage technologies. What if we provided an Explorer specifically targeted at this type of customer that provided the customization, personalization, and workflow tuning that would put us on a path to meeting their needs? While this is a numerically small set of customers, there is an opportunity to compete with Linux and move the influential enthusiast to feel more strongly about Windows.

As we think about working on various form factors we need to think about how we reconcile experiences which have been the result of parallel evolution. The most obvious example of this is media organization and playback. While we made significant steps in Windows 7 in the unification of Media Center and Media Player, as a company we continue to develop and diverge with Zune and Windows Mobile devices and services. This is one example of course, but one that we are all acutely aware of and simply must be addressed in Windows 8. We must look at how we reconcile these silos to make sure we do not solve them either superficially at the surface or claim that because we share some plumbing we are unified. We won’t be successful if we continue to release Windows with multiple ways of showing

## Microsoft – Windows Team Only

off the same scenario when it is one of the most visible and mainstream competitive scenarios. We should clarify these implementations while recognizing that Media Center represents our entrée to the television screen and only if we continue to improve the access Windows customers have to television can we realize our complete vision for all “three screens”. We should look to the path we took with Email, Calendar, Photos, and Video with Windows Live and recognize we can make tough choices and focus our energies in a single place and do a better job.

We, of course, assume that all of our form factors rely on a web connection. This starts with Windows Update/Microsoft Update as the core services we offer to keep a form factor up to date. This continues as we include the Windows web presence which is the cornerstone of how consumers learn about Windows, learn to use Windows, and connect with members of a community to share and grow together in their use of Windows through tutorials, documentation, and a wide variety of "tools" that help in their use of Windows. As we plan Windows 8, the obvious next step is to connect this community with a "store" of software. This should be a forward-looking approach and one that is driven from where we see innovation in the platform and how we can build on Windows Browsing, Graphics, and so on to deliver a *well-defined* application experience for a broad set or *class* of applications.

**Windows Live** represents our implementation of software + services provided to all Windows end-users and developers—this goes beyond “consumers” and represents an audience equivalent to the Windows audience. We’ve started from a disparate set of services, client code that did not connect to our services, and a nascent platform and with Wave 3 and now with the plans for Wave 4 coming together we are gaining momentum that goes beyond the nearly 500,000,000 people that use our individual services today to recognition of the asset we are building. The Vision for Wave 4 will detail the competitive landscape and the work that we need to do to make sure that we continue to broaden the usage and appeal of the software + services we are delivering. As we think of Windows Live in the context of Windows 8 we want to consider how Windows Live helps us to accomplish two main initiatives (1) Windows Live enhances, but does not replace or duplicate anything in Windows and (2) Windows Live provides the platform for developers to connect to Windows Live services and to also provide synchronization and connection to service back-ends to Windows-based software.

We will expect Windows Live to top off the **Modern Windows Experience** particularly as it relates to consumer scenarios around photos, video, email, calendar, information sharing, and collaboration. Wave 4 will expand this to include opportunities for Office to connect to this infrastructure and expose the functionality of Office 14’s web-based productivity and also connect these services to the Office client applications.

We also expect to use the platform capabilities of Wave 4 to begin to offer customers solutions to some of the most frustrating aspects of owning multiple PCs or using multiple PCs—access to data, roaming commonly used settings, or keeping files across multiple PCs in sync.

An example that brings together the conceptual framework as well as our desire to build great applications for Windows across form factors, while doing so using web technologies, is to think about the Live Search phone application (search, maps, movies, etc.) Today this application is only available on

## Microsoft – Windows Team Only

Windows Mobile, yet clearly would be really cool to use on Windows. And at the same time it represents essentially a front end to web services all of which are rendered in a browser today. While there are clearly complexities in the “rendering” of a user interface for different screen sizes, the notion that building a great application like this, using Windows-based tools, developing in web-centric technologies, while being able to have the quality of interaction one associates with a “client” is both compelling and necessary. Then imagine a vast library of such applications available as part of a “store” all following this “app model”.

There are many elements of Windows that are part of developing the overall Windows experience: SDKs, diagnostic tools, qualification tools and logos, upgrade advisors, etc. A significant part of planning Windows 8 will be doing more to bring these together to be part of our single effort at delivering the product, where we align goals, deliverables, schedules, user experience, and so on so that all the elements that are part of the Windows platform are engineered to the same standards, delivered with the same integrity, and viewed by customers as part of a complete end-to-end system.

Looking at this conceptual view of Windows 8 and the complete Windows experience provides some indication of the breadth and depth required to plan Windows 8. The examples and scenarios described here are not meant to be complete, but to illustrate the fantastic technology base and great opportunities we have ahead of us. It is important to recognize as fast we sometimes think things are changing, well-designed products take time to develop and the universe of 1,000,000,000 PC users will not change over-night. If we are thoughtful, planful, and execute well we can build on this conceptual framework and make Windows 8 a *well-executed and breakthrough* release for customers of all types, all around the world.

## Next Steps and Timeline

This is the first step in planning Windows 8. Planning a product such as Windows is a significant undertaking. We will come together as “one Windows” to plan the release, working with partners across the company. Our job as a team is to understand the context in which we are building Windows, bring new ideas to the table, learn from each other, listen to all points of view, and develop a consensus for the themes, scenarios, and features of Windows 8—a plan that embodies the success characteristics in this memo.

We know that we’re all going to start working hard to develop ideas and approaches for Windows 8. Let’s not lose sight of our desire to improve on how we worked for Windows 7. We have a few activities that we will be kicking off in the near term, building on the process outlined recently in the internal blog post [Participatory Planning and the Next Release](#).

1. **Engineering System Improvements.** As we finalize Windows 7 we will of course take a step back to gather feedback and learn what we should improve as part of the overall engineering system. We will take a structured approach to improvements and treat improving the engineering system in a manner quite similar to developing the plans and executing the product cycle. Improving the engineering system is a precursor to the start of the schedule for Windows

8 and so it will start soon. We will look at tools and processes that most need to improve and also to reinforce our commitment to Common Engineering Criteria as part of how we develop Server roles. To kick off planning the improvements we will make, a memo outlining the goals and next steps for improving the system will follow shortly.

2. **Planning for MQ.** MQ is treated exactly like any other milestone. The work we do must follow a clear “spec” with the right resources for success, and defined exit criteria. MQ is not a milestone for unplanned or unspecified feature work nor is it a milestone to fix a lot of bugs without adequate test coverage. MQ is a milestone that is about significantly improving the base quality of Windows and the robustness of our engineering system. We will reaffirm the ownership of components and begin to lay the groundwork for any functionality we plan to deprecate. Our test leaders will lead us through MQ as we conclude Windows 7.
3. **Learning from Windows 7.** As a learning organization we will expect to spend a lot of time learning across the team about what has gone well and what needs to improve. This will focus on the engineering and management processes we have used. This *post-mortem* work is a big part of bringing closure to the release and to making sure we have a specific set of improvement we will make in developing Windows 8.
4. **Resource allocation and alignment.** We have engineering “philosophy” in place and we’ll continue along these same lines—discipline leadership, with the operation of our efforts using right-sized feature team triads. But we all know as part of Windows 7 that some things were not perfectly aligned organizationally and we know where we have areas that would benefit from some organizational tweaking. A part of these changes will be for each member of the team to consider new opportunities as well—if you’ve worked on the same technology for several release cycles it might make sense (but is not necessary or required) to look around (and by contrast if you have not, then it might not make too much sense to look around). Your manager is always the best place to start these conversations. This notion of career management is based on balancing depth of knowledge in the team with broadening a person’s experience to develop superior judgment. It is not intended to “make” people move if they are very happy in their depth technical area, but we do ask people to consider their own long term goals. There will be changes, but these changes are going to be deliberate and related to the work we’re trying to do. This is an optimization that is a routine part of building a great team for the next release. Along with feedback from Windows 7, the framing for the release informs any adjustments we will make to the organization as well. Obviously we can’t have a complete plan without teams knowing the resource and organizational adjustments, and of course we want to have some information about planning to execute on these. The structure we have in place also allows us a great deal of flexibility over how work gets done once we begin and we will trust that to take place.

Probably the most common question at this point comes from folks who have ideas and want to know “how do I get my idea into Windows”. That’s a good question, but in a sense it is important to take a step back. The process of planning is about doing everything we can to develop a *high integrity* plan—that is a plan that holds together as one, meets the needs of customers and business, is achievable from an execution perspective. To achieve this requires that everyone participate in planning. It won’t be

## Microsoft – Windows Team Only

enough to submit a suggestion, nor will it be appropriate to get a feature done and then use the process of planning to see how to squeeze it in. Creating the Vision is a process of iteration and refinement. Creating a plan is about “editing” the set of possible ideas by looking constantly at what is achievable and what continues to reinforce the overall goals we set out to achieve. The best way to see any specific idea become part of Windows 8 is to participate while also recognizing that everyone shares the same goal of getting to the best set of features that make the most sense for the release. We have gone through this process as a team and we can improve how we all work together to achieve the plan. Two blog posts that illustrate both the challenges and opportunities worth referring to include [Nice plan, but not enough...](#) and [From brainstorming to plan to feature—selling your ideas](#). Planning is hard. The best way to make it easy is to take advantage of the opportunities to participate.

It is important to call out that during planning we will also be actively inviting the participation of our partner groups—across developer division, entertainment and devices, Office, and of course Server. Our commitment is to be extremely inclusive and in return we expect a commitment to the spirit of participating in planning.

As we look to the overall timeline and rhythm for planning, program management will take the lead in developing an overall planning calendar. We are not going to announce a full project schedule, but rather frame the schedule with this memo. We have committed to customers, enterprise and OEM alike, that we won’t stretch out the time between releases beyond about 3 years. We intend to stick to this. We also intend to allow adequate time to develop a very strong plan such that the execution of Windows 8 can be an improvement over the excellent execution of Windows 7. The timeline below assumes a July RTM with availability in the fall for Windows 7.

Time	Execution Efforts
<b>May 2009</b>	Framing memo Engineering kickoff and memo
<b>June 2009</b>	Resource allocation and alignment
<b>June – August 2009</b>	Planning efforts such as offsites, brownbags, and product planning sharing of research begin.
<b>August 2009</b>	Planning Forum with broad participation from Windows and partner teams Engineering post-mortem process to learn from Windows 7 and improve the engineering of Windows 8
<b>September 2009</b>	Planning memo with specific themes and outlining the framework and goals of the release.
<b>October – January 2009</b>	Specific planning efforts focused on themes and scenarios.
<b>February 2010</b>	Vision with prototypes complete. Includes rollout to teams and partners.
<b>March 2010 – Spring/Summer 2012</b>	Complete specs, develop and Execute on the schedule <i>Note: Windows 8 RTMs within 3 years of Windows 7 RTM</i>