The Internet Services Disruption

To:	Executive Staff and direct reports
Date:	October 28, 2005
From:	Ray Ozzie
Subject:	The Internet Services Disruption

It is an exciting time, as we're at the beginning of the biggest product cycle in the company's history. In a week we ship new versions of Visual Studio, SQL Server and BizTalk Server. Later this month we ship Xbox 360. Next year we have a double barreled release of our two largest products with Windows Vista and Office "12". It's a great time for customers, our partners, and for those at Microsoft who have put so much of themselves into these products.

But we bring these innovations to market at a time of great turbulence and potential change in the industry. This isn't the first time of such great change: we've needed to reflect upon our core strategy and direction just about every five years. Such changes are inevitable because of the progressive and dramatic evolution of computing and communications technology, because of resultant changes in how our customers use and apply that technology, and because of the continuous emergence of competitors with new approaches and perspectives.

In 1990, there was actually a question about whether the graphical user interface had merit. Apple amongst others valiantly tried to convince the market of the GUI's broad benefits, but the non-GUI Lotus 1-2-3 and WordPerfect had significant momentum. But Microsoft recognized the GUI's transformative potential, and committed the organization to pursuit of the dream – through investment in applications, platform and tools – based on a belief that the GUI would dramatically expand and democratize computing.

When we reflected upon our dreams just five years later in 1995, the impetus for our new center of gravity came from the then-nascent web. With a clear view upon the challenges and opportunities it presented, the entire company pivoted to focus on the internet to pursue that 'fully connected' dream with support for internet standards throughout our product line: a web browser, server and development tools, and a service in MSN that was transformed into a web portal. Many things we developed in that era continue to fuel the growth of today's internet: the technologies of AJAX – DHTML and XMLHTTP – were created in 1998 and used in products such as OWA.

In 2000, in the waning days of the dot com bubble, we yet again reflected on our strategy and refined our direction. After taking a more deliberative look at the internet and its implications for software, we came to the conclusion that the internet would go beyond browsing and should support programmability on a global scale. We observed that certain aspects of our most fundamental platform – the tools and services that developers use when building their software – would not likely satisfy the emerging security and interoperability requirements of the internet. So we embarked upon .NET, a transformative new generation of the platform and tools built around managed code, the XML format and web services programming model. At the time, it was a risky bet to build natively around XML, but this bet paid off handsomely and .NET has become the most popular development environment in the world.

It is now 2005, and the environment has changed yet again – this time around *services*. Computing and communications technologies have dramatically and progressively improved to enable the viability of a services-based model. The ubiquity of broadband and wireless networking has changed the nature of how people interact, and they're increasingly drawn toward the simplicity of services and service-enabled software that 'just works'. Businesses are increasingly considering what services-based economics of scale might do to help them reduce

infrastructure costs or deploy solutions as-needed and on subscription basis.

Most challenging *and promising* to our business, though, is that a new business model has emerged in the form of advertising-supported services and software. This model has the potential to fundamentally impact how we and other developers build, deliver, and monetize innovations. No one yet knows what kind of software and in which markets this model will be embraced, and there is tremendous revenue potential in those where it ultimately is.

Just as in the past, we must reflect upon what's going on around us, and reflect upon our strengths, weaknesses and industry leadership responsibilities, and respond. As much as ever, it's clear that if we fail to do so, our business as we know it is at risk. We must respond quickly and decisively.

The Landscape

Since 1995, inexpensive computing and communications technologies have advanced at a rapid rate that even exceeded our expectations. It's so very difficult now for us to imagine a world without the PC, the web and the cell phone. In the US, there are more than 100MM broadband users, 190MM mobile phone subscribers, and WiFi networks blanket the urban landscape. This pattern is mirrored in much of the developed world. Computing has become linked to the communications network; when a PC is purchased, it's assumed that the PC will have high-speed internet connectivity. At work, at home, in a hotel, at school or in a coffee shop, the networked laptop has become our 'virtual office' where we file our information and interact with others. The broad accessibility and rapid pace of innovation in hardware, networks, software and services has catalyzed a virtuous cycle whose pace isn't slowing. There has never been a more exciting time to be a developer or a user of technology.

Our products have embraced the internet in many amazing ways. We've transformed the desktop into a rich platform for interactive internet browsing, media and communications-centric applications. We've transformed Windows into best-of-breed infrastructure for internet applications and services. We've created, in .NET, the most popular development platform in the world. We've got amazing products in Office and our other IW offerings, having fully embraced standards such as XML, HTML, RSS and SIP. Our MSN team has demonstrated great innovation and has held its own in a highly competitive and rapidly changing environment – particularly with Spaces and in growing a base of 180M active Messenger users worldwide. The Xbox team has also built a huge user community and has demonstrated that internet-based "Live" interaction is a high-value, strong differentiator.

But for all our great progress, our efforts have not always led to the degree that perhaps they could have. We should've been leaders with all our web properties in harnessing the potential of AJAX, following our pioneering work in OWA. We knew search would be important, but through Google's focus they've gained a tremendously strong position. RSS is the internet's answer to the notification scenarios we've discussed and worked on for some time, and is filling a role as 'the UNIX pipe of the internet' as people use it to connect data and systems in unanticipated ways. For all its tremendous innovation and its embracing of HTML and XML, Office is not yet the source of key web data formats – surely not to the level of PDF. While we've led with great capabilities in Messenger & Communicator, it was Skype, not us, who made VoIP broadly popular and created a new category. We have long understood the importance of mobile messaging scenarios and have made significant investment in device software, yet only now are we surpassing the Blackberry.

And while we continue to make good progress on these many fronts, a set of very strong and determined competitors is laser-focused on internet services and service-enabled software. Google

is obviously the most visible here, although given the hype level it is difficult to ascertain which of their myriad initiatives are simply adjuncts intended to drive scale for their advertising business, or which might ultimately grow to substantively challenge our offerings. Although Yahoo also has significant communications assets that combine software and services, they are more of a media company and – with the notable exception of their advertising platform – they seem to be utilizing their platform capabilities largely as an internal asset. The same is true of Apple, which has done an enviable job integrating hardware, software and services into a seamless experience with dotMac, iPod and iTunes, but seems less focused on enabling developers to build substantial products and businesses.

Even beyond our large competitors, tremendous software-and-services activity is occurring within startups and at the grassroots level. Only a few years ago I'd have pointed to the Weblog and the Wiki as significant emerging trends; by now they're mainstream and have moved into the enterprise. Flickr and others have done innovative work around community sharing and tagging based on simple data formats and metadata. GoToMyPC and GoToMeeting are very popular low-end solutions to remote PC access and online meetings. A number of startups have built interesting solutions for cross-device file and remote media access. VoIP seems on the verge of exploding – not just in Skype, but also as indicated by things such as the Asterisk soft-PBX. Innovations abound from small developers – from RAD frameworks to lightweight project management services and solutions.

Many startups treat the 'raw' internet as their platform. At the grassroots level, such projects actively use standards such as vCards and iCal for sharing contacts and calendars. Most all use RSS in one way or another for data sharing. Remixing and mashing of multiple web applications using XML, REST and WS is common; interesting mash-ups range from combining maps with apartment listings, to others that place RSS feeds on top of systems and data

not originally intended for remixing. Developers needing tools and libraries to do their work just search the internet, download, develop & integrate, deploy, refine. Speed, simplicity and loose coupling are paramount.

And the work of these startups could be improved with a 'services platform'. Ironically, the same things that enable and catalyze rapid innovation can also be constraints to their success. Many hard problems are often ignored – the most significant of which is achieving scale. Some scale issues are technological and result from the fact that they are generally built on application server platforms rather than high-scale service platforms. But new services also need to build user communities from scratch – generally by word of mouth. Many fund their sites using syndicated ads, but have a difficult time transforming their services into higher levels of commerce. Some seek to incorporate client software into their user experience, but then need to reinvent software deployment, update, communications and synchronization mechanisms. User identity and cross-service interoperability mechanisms are still needlessly fragmented. Intuitively there seems to be a platform opportunity in providing such capabilities to developers in a form that retains the speed, simplicity and loose coupling that is so very important for rapid innovation.

Key Tenets

Today there are three key tenets that are driving fundamental shifts in the landscape – all of which are related in some way to services. It's key to embrace these tenets within the context of our products and services.

1. *The power of the advertising-supported economic model.*

Online advertising has emerged as a significant new means by which to directly and indirectly fund the creation and delivery of software and services. In some cases, it may be possible for one to

obtain more revenue through the advertising model than through a traditional licensing model. Only in its earliest stages, no one yet knows the limits of what categories of hardware, software and services, in what markets, will ultimately be funded through this model. And no one yet knows how much of the world's online advertising revenues should or will flow to large software and service providers, medium sized or tail providers, or even users themselves.

2. The effectiveness of a new delivery and adoption model.

A grassroots technology adoption pattern has emerged on the internet largely in parallel to the classic methods of selling software to the enterprise. Products are now discovered through a combination of blogs, search keyword-based advertising, online product marketing and word-of-mouth. It's now expected that anything discovered can be sampled and experienced through self-service exploration and download. This is true not just for consumer products: even enterprise products now more often than not enter an organization through the internet-based research and trial of a business unit that understands a product's value.

Limited trial use, ad-monetized or free reduced-function use, subscription-based use, on-line activation, digital license management, automatic update, and other such concepts are now entering the vocabulary of any developer building products that wish to successfully utilize the web as a channel. Products must now embrace a "discover, learn, try, buy, recommend" cycle – sometimes with one of those phases being free, another ad-supported, and yet another being subscription-based. Grassroots adoption requires an end-to-end perspective related to product design. Products must be easily understood by the user upon trial, and useful out-of-the-box with little or no configuration or administrative intervention.

But enabling grassroots adoption is not just a product design issue. Today's web is fundamentally a self-service environment,

and it is critical to design websites and product ‘landing pages’ with sophisticated closed-loop measurement and feedback systems. Even startups use such techniques in conjunction with pay-per-click advertisements. This ensures that the most effective website designs will be selected to attract discovery of products and services, help in research and learning, facilitate download, trial and purchase, and to enable individuals’ self-help and making recommendations to others. Such systems can recognize and take advantage of opportunities to up-sell and cross-sell products to individuals, workgroups and businesses, and also act as a lead generation front-end for our sales force and for our partners.

3. The demand for compelling, integrated user experiences that “just work”.

The PC has morphed into new form factors and new roles, and we increasingly have more than one in our lives – at work, at home, laptops, tablets, even in the living room. Cell phones have become ubiquitous. There are a myriad of handheld devices. Set-top boxes, PVRs and game consoles are changing what and how we watch television. Photos, music and voice communications are all rapidly going digital and being driven by software. Automobiles are on a path to become smart and connected. The emergence of the digital lifestyle that utilizes all these technologies is changing how we learn, play games, watch TV, communicate with friends and family, listen to music and share memories.

But the power of technology also brings with it a cost. For all the success of individual technologies, the array of technology in a person’s life can be daunting. Increasingly, individuals choose products and services that are highly-personalized, focused on the end-to-end experience delivered by that technology. Products must deliver a seamless experience, one in which all the technology in your life ‘just works’ and can work together, on your behalf, under your control. This means designs centered on an intentional fusion of internet-based services with software, and sometimes even hardware, to deliver meaningful experiences and solutions with a

level of seamless design and use that couldn't be achieved without such a holistic approach.

The Opportunities

These three tenets are causing a shift in the software landscape that started with consumers and is progressively working its way toward the enterprise – changing how software is monetized, how software is delivered, and what kind of software is ultimately embraced. With our presence in so many markets serving so many audiences, and with such a broad variety of products and solutions, we are well positioned to deliver *seamless experiences* to customers, enabled by services and *service-enhanced software*, including:

SEAMLESS OS – The operating system as it would be designed for today's multi-PC, multi-device, work anywhere, web-based world. Enabling you to login using any of your service-based or enterprise identities. Deploying software automatically and as appropriate to all your devices, and roaming application data and settings. Permitting seamless access to storage across all your PCs, devices, servers and the web.

SEAMLESS COMMUNICATIONS – Communications and notifications – from voice to typing to shared screen; from PC to service-based agent to phone. Maintaining continuous co-presence with intimate friends and family; improving the coordination amongst individuals who need to work together by reducing latency and adding clarity through shared context.

SEAMLESS PRODUCTIVITY – Enabling you to create, find and organize documents and data among all the desktops, devices, servers and services to which you have access, and with all the others with whom you need to work, through 'shared space' products that are internet service-based, enterprise server-based

and directly peer-to-peer. Working within and across homes, small businesses, virtual workgroups and enterprises.

SEAMLESS ENTERTAINMENT – Enabling you to create, store, organize, present, consume and interact with media of all kinds; accessing, caching and viewing it anywhere you like regardless of where the media resides. Gaming experiences that bring two or two million people together across PCs, devices and the web.

SEAMLESS MARKETPLACE – Enabling you to research, find, buy and sell whatever you want through a seamlessly integrated purchase, billing & payment & points, advertising & lead generation & sales management system designed to satisfy the needs of both buyers and sellers.

SEAMLESS SOLUTIONS – Enabling workgroups and businesses to rapidly create and customize any of a broad class of template-driven, semi-structured data-based applications and solutions that "just work" and provide instant value – whether using them from the web, from enterprise servers, or from mobile client PCs.

SEAMLESS IT – Enabling enterprises to seamlessly and cost-effectively manage many of the things they've classically done within their data centers – e.g. PCs, messaging, content and applications. The management experience might be wholly within the cloud, or with the cloud seamlessly integrating enterprise server assist.

Moving Forward

In order to adapt to the requirements underlying these key tenets, groups must reflect upon their existing plans, and assess their designs in the context of the end-to-end experiences they need deliver in order to understand how services might make a substantive impact. Groups should consider how new delivery and

adoption models might impact plans, and whether embracing new advertising-supported revenue models might be market-relevant.

In assessing where we are and where we need to be, some new efforts will surely require incubation. But in many areas we have 80% of the product and technical infrastructure already built – we just need to close the 20% gap. Following are but a few thoughts for each division intended to catalyze a "services-enhanced software" mindset.

Platform Products & Services Division

a. BASE vs. ADDITIVE EXPERIENCES

In MSN, and in Windows Update and software deployed by it, we have quite a bit of experience with methods and practices for getting innovations to market on a rapid cycle. In the form of a newly combined division, we should consider many options as to how we might bring *user experience* innovations and enhancements to users worldwide. Specifically, we should consider the achievability, desirability, and methods of increasing the tempo for both 'base' OS experiences as well as 'additive' experiences that might be delivered on a more rapid tempo. In doing so, we would better serve a broad range of highly-influential early adopters.

b. SERVICES PLATFORM

Through years of experience, the MSN team understands the methods and practices of building 'internet scale' services. The Platform team understands developers and has deep experience in communications and storage architectures. These teams must work together, benefiting from each others' strengths, to develop a next generation internet services platform – a platform for both internal and external innovation. A platform with capabilities and an operations infrastructure that takes those services to a scale never yet seen on the internet – to our benefit, and to the benefit of our partners and customers.

c. SERVICE/SERVER SYNERGY

A tension has emerged between our products designed for the enterprise and those for the internet. Exchange/Hotmail, AD/Passport, and Messenger/Communicator are but three examples. All our enterprise clients and servers must interoperate with and complement our internet services. Our functional aspirations are generally “server/service symmetry”, but architectural considerations dictate that different implementations may be required to economically reach internet scale. We must quickly find the best path to achieve seamless user, developer, and administration experiences involving servers and services.

d. LIGHTWEIGHT DEVELOPMENT

The rapid growth of application assembly using things such as REST, JavaScript and PHP suggests that many developers gravitate toward very rapid, lightweight ways to create and compose solutions. We have always appreciated the need for lightweight development by power users in the form of products such as Access and SharePoint. We should revisit whether we’re adequately serving the lightweight model of development and solution composition for all classes of development.

e. RESPONSIBLE COMPETITION

We will compete energetically but also responsibly and with recognition of our high legal responsibilities. We will design and license Windows and our internet-based services as separate products, so customers can choose Windows with or without Microsoft’s services. We’ll design and license Windows and our services on terms that provide third parties with the same ability to benefit from the Windows platform that Microsoft’s services enjoy. Our services innovations will include tight integration with the Windows client via documented interfaces, so that competing services can plug into Windows in the same manner as Microsoft’s services. We will compete hard and responsibly in services on the

basis of software innovation and price – and on that basis we will offer consumers and businesses the best value in the market.

Business Division

a. CONNECTED OFFICE

How would we extend or re-conceptualize Office modules to fit in this seamless model of connectedness to others, and to other applications? Should PowerPoint directly 'broadcast to the web', or let the audience take notes and respond? How should we increase the role of Office Online as the portal for productivity? What should we do to bring Office's classic COM-based publish-and-subscribe capabilities to a world where RSS and XML have become the de facto publish-and-subscribe mechanisms?

b. TELECOM TRANSFORMATION

How should our investments in RTC evolve to serve not just the enterprise, but also fully embrace the concept of grassroots adoption? How can RTC begin as an individual phenomenon, growing into a small business offering with a level of function that they'd never imagine possible, growing into the enterprise? How should we utilize service-based federation and hosting to ensure a 'just works' experience for all users, whether or not an administrator was ever involved?

c. RAPID SOLUTIONS

How can we utilize our extant products and our knowledge of the broad historical adoption of forms-based applications to jump-start an effort that could dramatically surpass offerings from Quickbase to Salesforce.com? How could we build it to scale to hundreds of millions of users at an unimaginably low cost that would change the game? How could we re-shape our client-side software offerings such as Access and Groove, and our server offerings such as SharePoint, to grow and thrive in the presence of such a

service? Could these rapid solutions encourage a new ISV ecosystem and business model?

Entertainment & Devices Division

a. CONNECTED ENTERTAINMENT

How can XBox Live benefit from interconnection with other services assets, such as PC-based and mobile-based IM and VoIP? How might both the PC and XBox mutually benefit from a common marketplace? Might PC users act as spectators/participants in XBox games, and vice-versa?

b. GRASSROOTS MOBILE SERVICES

How might the Windows Mobile device experience be transformed by for consumers by connection to a services infrastructure – in particular one enabled by RTC-based unified communications? How might unmediated connection to a rich services infrastructure transform mobile phones into a mass market messaging, media and commerce phenomenon?

c. DEVICE/SERVICE FUSION

What new devices might emerge if we envision hardware/software/service fusion? What new kinds of devices might be enabled by the presence of a service?

What's Different?

One perspective on this memo might be to say "This is in many ways is pretty close to what we're already working on. What's the big deal?" Or "We tried something similar years ago; why will we succeed this time?" These are understandable reactions. Many visions of the future going all the way back to "Information at Your Fingertips" contain elements of what has been laid out here.

That said, I have a number of reasons for optimism that we can deliver well on this vision. First, I know that Bill, Steve and the senior leadership team understand that Microsoft's execution effectiveness will be improved by eliminating obstacles to developing and shipping products. The recent reorganization into three divisions is a significant step, and the division presidents are committed to changes to improve our agility.

Second, we are just now completing a wave of innovation that has never been seen in this company. 2006 is going to be an amazing year for shipping products, and many across the company will be ready to take on a new mission.

Third, regardless of past aspirations, this is the right time to be focusing on services for two specific reasons: the increasing ubiquity of broadband has made it viable, and the proven economics of the advertising model has made it profitable. It can be argued, for example, whether or not Hailstorm was the 'right' undertaking. But regardless, the effort would certainly have benefited from having a known-viable services business model for which to design.

Finally, I believe at this juncture it's generally very clear to each of us why we need to transform – the competitors, the challenges, and the opportunities. As an outsider, I was repeatedly impressed and awed over the years by how this company's talent has swarmed to effectively respond to huge business challenges and transitions.

That said, even when we've been solidly in pursuit of a common vision, our end-to-end execution of key scenarios has often been uneven – in large part because of the complexity of doing such substantial undertakings. In any large project, the sheer number of moving parts sometimes naturally causes compartmentalization of decisions and execution. Some groups might lose sight of how their piece fits in, or worse, might develop features without a clear understanding of how they'll be used. In some cases by the time the vision is delivered, the pieces might not quite fit into the

originally-envisioned coherent whole. We cannot allow the seams in our organization, or our methods of making decisions, show through in our products, or result in the failure to deliver on key end-to-end experiences.

Complexity kills. It sucks the life out of developers, it makes products difficult to plan, build and test, it introduces security challenges, and it causes end-user and administrator frustration. Moving forward, within all parts of the organization, each of us should ask "What's different?", and explore and embrace techniques to reduce complexity.

Some problems are inherently complex; there is surely no silver bullet to reducing complexity in extant systems. But when tackling new problems, I've found it useful to dip into a toolbox of simplification approaches and methods. One such tool is the use of extensive end-to-end *scenario-based design* and implementation. Another is that of utilizing loosely-coupled design of systems by introducing constraints at key junctures – using standards as a tool to force quick agreement on interfaces. Many such tools are not rocket science: for example, by forcing a change in practices to increase the frequency of release cycles, scope and complexity of any given release by necessity is greatly reduced. Another simple tool I've used involves attracting developers to use common physical workspaces to naturally catalyze ad hoc face-time between those who need to coordinate, rather than relying solely upon meetings and streams of email and document reviews for such interaction. Embracing change at a local level through such tools can make a real difference – one project at a time.

Next Steps

We're off to a great start with many initiatives already under way – from efforts occurring now within MSN, to the IW services being launched imminently. We're in a tremendous position to succeed,

but doing so will require your belief, creativity, support, leadership, follower-ship and action.

This memo was intended to get all of us roughly on the same page, and to get you thinking. The next steps are:

1) I am working with the division presidents to assign, by December 15th, "scenario owners" – a role intended to improve our execution of key services-based initiatives through leadership. These leaders will provide an outside-in perspective in mapping out and communicating specific market objectives, while at the same time working with developers and others at the detail level to ensure expedient decision making and continuity. These individuals will be responsible for driving critical decisions such as feature re-prioritization and cuts while appreciating the business tradeoffs and impact of such decisions. They'll listen. They'll rapidly effect changes in plans to ensure execution and improve agility, even for scenarios that span divisions. Initial scenarios to be assigned ownership will include the seven seamless experiences described earlier.

2) Beginning in January these individuals will work with me and with product groups to concretely map out scenarios and pragmatically assess changes needed in product and go-to-market plans related to services and service-based scenarios. For some groups this will impact short-term plans; for many others on path to shipping soon, it will factor significantly into planning for future releases.

3) All Business Groups have been asked to develop their plans to embrace this mission and create new service offerings that deliver value to customers and utilize the platform capabilities that we have today and are building for the future. We expect both technical and non-technical communities to be increasingly engaged on the topic of services and service-enhanced software. As we begin planning the next waves of innovation –

such as those beyond Vista and Office “12” – we will mobilize execution around those plans.

4) I have created an internal blog that will be used to notify you of further plans as they emerge. There, I’ll point you to libraries of documents that you will find interesting to read, and I’ll be experimenting with ways that you can directly engage in the conversation.
http://blogs/live

These steps are important and necessary, but not sufficient, for us to deliver on our aspirations. The most important step is for each of us to internalize the transformative and disruptive potential of services. We must then focus on the need for agility in execution, and take actions as appropriate where each of us can.

The opportunities to deliver greater value to our customers, to our developer and partner communities, and to our shareholders are significant. I very much look forward to embarking on this journey with all of you.

— Ray