Microsoft .NET in Telecommunications

How Web Services Can Help Telecom Companies Reduce Costs, Increase Revenues, and Decrease Response Time

Verizon Communications is saving $10 million per year in hardware costs alone by moving its call center and customer billing system from mainframes to a Web-based system. The solution also significantly decreases the time required to make a functional or process change—it now takes one-fifth to one-tenth the time previously required.

Broadwing, an Ohio-based CLEC (Competitive Local Exchange Carrier), created a Web-based DSL provisioning system that boosts operations staff productivity by 300 percent, dramatically accelerates DSL ROI, and increases customer satisfaction. The new solution integrates five separate systems, allowing fully automated provisioning in over 95 percent of cases.

T-Mobile, the mobile phone subsidiary of Deutsche Telekom, created a Web-based mobile service platform that is now driving new revenue streams. The platform, which connects mobile users with their business applications, is highly scalable and reliable—and it took less than a year to create.

These diverse stories have two things in common.

First, they illustrate the emerging trend in the telecom industry of applying Web services technology to reduce costs, increase revenues, and decrease response times. Telecom companies that take advantage of this trend are gaining unprecedented flexibility, which helps them to survive—and even thrive—in a difficult market.

Second, Microsoft® .NET—the software that connects information, people, systems, and devices—enabled each of these companies to create an economical solution that gives them the agility they need to adapt to changing conditions.

This paper describes:

- The challenges faced by telecom companies as they struggle with both economic problems and the difficulties of integrating disparate systems
- How Web services can help telecom companies become more agile, better connected, and more profitable
- The opportunities offered by Microsoft .NET-connected software and the advantages of .NET-enabled solutions over other Web services approaches—together with real-world examples of how telecom companies are using .NET-connected solutions today to decrease costs and increase revenues
- A set of concrete steps that telecom companies can take to move toward .NET connectivity

Telecom Industry Challenges

The telecom industry consists of many types of companies, including Baby Bells, CLECs, wireless carriers, ISPs, and Web hosting companies. While each component of theindustry has its own individual challenges, certain economic and technological challenges apply to the industry as a whole.

On the economic side, most telecom companies have been experiencing declining margins. One reason is increased competition and a glut of backhaul lines. In this environment, the only way companies are able to retain and grow their subscriber base and Average Revenue Per User (ARPU) and drive increased network usage is to slash prices. The flip side of this coin is that the value-added services that telecom companies expected to generate increased revenues so far have not materialized. As a result of these economic difficulties, the US telecom industry reduced its job count by 20 percent in 2001.

On the technology side, the inability of different systems within a company (billing, customer care, provisioning, trouble ticket, and so forth) to communicate with each other creates inefficiencies, making it difficult for companies to reduce costs. These incompatibilities between systems also make it hard for employees to access the information they need; increase training costs and time for new employees; and make it complicated and costly to add new products, since the new products have to be linked to each of these disparate systems.

System incompatibility problems are further exacerbated by recent industry-wide consolidation, acquisitions, and bankruptcies. Recent examples include the acquisition of Exodus by Cable & Wireless and the bankruptcy declarations of Worldcom and Excite@home. These mergers and acquisitions result in large numbers of incompatible IT systems. For example, one North American telecom company currently has eight different billing systems as a result of mergers and acquisitions.

The economic problems faced by the telecom industry are resulting in shrinking margins, which continue to squeeze profits. While circuit voice is still generating significant revenues, growth in that area is flat. Future opportunity for growth lies primarily in the mobile, data, and packet voice sectors.

Telecom companies have already cut their capital expenditures as much as possible. Now, the best way for them to stay profitable and grow is to:

1. **Lower the cost of delivering and supporting current services.** Integrating systems and simplifying the process of changing them will significantly reduce the cost of operational and business support systems. If this integration and simplification can be accomplished by leaving existing systems in place and layering new technologies on top of them, rather than “ripping and replacing” legacy systems, telecom companies can realize both cost savings and increased agility, making them better able to respond rapidly to changing business conditions. And they can do so without losing their investment in their existing systems.

2. **Increase revenues and grow market share by offering differentiated services.** This step, in turn, has three parts:

    - **Developing new functionality for existing products** to drive increased usage, resulting both in greater use of network capacity and higher ARPU. For example, adding an alert capability to an existing voicemail service—one that would pop up a message on the user's desktop PC, PDA, or Smartphone when a voicemail was received on any of the user's phone numbers—could result in greater usage of the voicemail service, generating increased revenues.

    - **Launching compelling new services** that will further increase data traffic and ARPU. Since any new services must be connected to existing operational and business systems, the integration and simplification process described in Step 1 will play a key role in making it possible to add such new services cost-effectively.

    - **Compressing development times** for new products by making existing networks and products accessible to the tools and skill sets of mass market developers. Such an approach will enable telecom companies to launch new products in less time and respond more rapidly to emerging customer needs.

In order to achieve the goals of reducing costs and increasing revenues and market share, telecom companies need to become more agile. That's because what defines success in a telecom strategy or solution today can change literally overnight. Some of the forces driving change are internal: optimizing processes to meet the needs of customers and partners, accommodating a merger or acquisition, and taking advantage of emerging channel or product opportunities. Others are external: for example, new regulations, customer expectations, or technology, or fundamental changes in the business climate.

Information Technology (IT) departments can play a key role in giving their companies the agility they need to accomplish their goals of lowering costs and increasing revenues and market share. For example, by integrating the company's systems (provisioning, billing, customer care, and so forth) so that they can share data, IT will make key information broadly available both to employees and to external partners, thereby increasing productivity, streamlining workflow, and leading to more effective policies and decisions. Integrating the company's systems will also position IT as a strategic asset to the business—a driver of business change and innovation instead of a barrier to change.

The Solution: Web Services

The easiest and most cost-effective way to implement the connectivity that telecom companies need to survive and thrive is through Web services. Web services layer new integrating technologies on top of existing systems, rather than requiring them to be ripped out and replaced. This approach to enterprise application integration will enable the telecom industry to reduce costs while offering compelling new capabilities and products that will increase revenues.

What exactly are Web services? They are interoperable services that enable applications to share data and functionality across programming languages, platforms, and devices.

Based on industry-standard protocols, Web services aren't bound to any particular language or platform. By enclosing existing systems in an XML (eXtended Markup Language) wrapper, Web services enable any Web-service-enabled application to interoperate with any other Web-service-enabled application. An application exposed as a Web service does not need any knowledge of the applications that use it. This means that services can be "loosely coupled," connected on the fly to create composite solutions tailored to specific individual or business needs.

Web services provide an extraordinary degree of flexibility for the telecom industry, helping them to bring new revenue-generating services to market rapidly. For example, Web services offer the wireless sector a great way to respond to regional trends for wireless coverage across multiple countries or regions (such as across all of Europe). Similarly, the growing usage of location-based services and local number portability (enabling a consumer to take a phone number to any location) provide opportunities for a Web services-based distributed model in which information can be accessed anywhere—for example, enabling cell phones users to access e-mail, buddy lists, and calendar information residing on their PCs.

By enabling companies to preserve and connect existing systems, Web services help them leverage their existing investments. Web services are highly scalable, scaling down to devices, up to large systems, and out by adding new machines. They can work across multiple organizations and broad geographies. And, because they have broad industry support, connections between organizations and industries can take place quickly and easily.

Microsoft's Web Services Strategy: Microsoft .NET

Microsoft .NET is Microsoft's product, services, and devices strategy for creating, deploying, managing, and consuming Web services. The products and services of the Microsoft platform provide an end-to-end solution for building, hosting, deploying, and using Web services, from the back office to the desktop. Built on XML, Microsoft .NET enables people to access information whenever and wherever they need it, without regard to the platform on which it runs, the application in which it resides, the programming language in which it is written, or the device they are using to access it.

.NET-connected software can help you (1) reduce your operating costs to increase profitability; (2) add new functionality to your current offerings to increase your revenues and drive network traffic; and (3) add new products that will help you achieve a "triple play"—that is, offer products that encompass all three telecom services: TV, phone, and broadband connectivity Furthermore, it allows you to do all this quickly and easily, both because of the great development tools available for the .NET platform and because .NET technology can be layered on top of your existing systems.

Reduce Your Operating Costs

With Microsoft .NET, you can integrate your Operations Support Systems (OSS) and Business Support Systems (BSS) services through a leave-and-layer strategy—that is, leaving existing systems in place and layering .NET-connected software on top of them.

In so doing, you will reduce your operating and business costs, optimize your assets, and make key information readily available throughout the enterprise.

For example, by using .NET technology to integrate your directory, provisioning, billing, rating, monitoring, customer care, and other line-of-business systems, you can increase operational agility, enabling these services to easily exchange information with each other. Such integration will also give employees better access to information, improving policy making and decision making. Similarly, you can use .NET technology to cost-effectively integrate your systems and business processes with those of a merged or acquired operation. Typically, .NET-based integration efforts take one-third the time and cost one-third as much as other enterprise application integration technologies.

.NET-connected software also offers a rapid, cost-effective means of upgrading your common business processes to respond to emerging trends, such as the increased usage of analytics, cross-customer information, and transaction-based billing.

For examples of how .NET systems can help telecom companies reduce their operating costs, consider the following three cases.

Example 1: Broadwing. At Broadwing, an Ohio-based CLEC, provisioning a DSL order involved five separate systems on different platforms with different data formats. Over 90 percent of the DSL orders required manual intervention, resulting in high operational costs. In addition, lost orders and a slow provisioning cycle led to low customer satisfaction.

Using Microsoft Visual Studio .NET, the Microsoft .NET Framework, and Microsoft BizTalk® Server, Broadwing was able to integrate the five systems and provide fully automated customer care, provisioning, and billing in over 95 percent of the cases. The new systems also increased the productivity of operations staff by 300 percent, dramatically accelerated DSL ROI, and increased customer satisfaction.

“We did this whole thing in just a matter of weeks,” said Alan Stone, Broadwing’s vice president of IT architecture and planning. “With another solution, it would have taken weeks just to figure out the most basic things.”

Example 2: Centerpost. Centerpost is an Illinois-based messaging and wireless service provider that works with clients like United Airlines, Travelocity.com, and The Weather Channel, sending messages to end users’ specified devices (for example, desktop PC, PDA, or cell phone) according to user-specified criteria. Because the messaging interface was proprietary, new clients often required assistance to integrate their existing systems with Centerpost’s service platform. The integration process typically took two to three engineers several months, costing the client upward of $100,000 in consulting fees.

Using Visual Studio.NET, Microsoft ASP.NET, and the .NET Framework, a single developer created a Web service interface in just three weeks—a productivity increase of at least 50 percent over their previous development environment. This interface enables Centerpost to expose its existing business logic for access over the Internet without having to modify it in any way. Now many clients can integrate their systems with the

company's service platform without any assistance at all. And those clients that do choose to use Centerpost's professional services experience a 75 percent reduction in consulting fees.

"We wanted to offer customers a single, standards-based interface into all the functionality we provide," says Craig Goren, Centerpost's founder and president. "By doing so, we reduce the time and effort required to integrate new clients, which lowers the barrier to their adopting our service and saves them significant up-front expenses. At the same time, it accelerates our ability to begin realizing new service revenues."

Add Functionality to Existing Products

With .NET-connected software, you can easily add new functionality to your existing products and integrate them in new ways. By doing so, you can deliver greater value to your customers and increase their use of your services. For example, by exposing your wireless data and network capabilities as mobile Web services, you could make integrated conferencing accessible across multiple devices, from PCs to Smartphones. As customers discover the increased ease of use and productivity of these integrated and broadly available services, they will be motivated to use them more frequently, thus driving increased ARPU. In addition, this increased usage will result in better use of your capacity, accelerating your return on investment.

Another example of new revenue-generating functionality that could be added to an existing product through .NET-connected software is a voicemail alert system. Such a system could pop up an alert on your desktop at work (or other device you designate) whenever you receive a voicemail at home. By adding value to your voicemail product, this new functionality would motivate additional customers to subscribe to your voicemail service, since the alert capabilities wouldn't work with an answering machine. In this way, the alert capabilities could drive additional revenues.

You could also use .NET-connected software to extend existing services to smart mobile clients, from Pocket PCs to Smartphones. .NET-enabled mobile devices are powerful, full-functioned devices that offer speed, graphics, memory, expandability, and the ability to integrate with PC applications. Based on Windows CE .NET, these devices expose the same underlying technologies and APIs as desktop systems, making it easy for developers to create applications that work on both mobile and PC platforms.

For example, telecom companies could offer mobile applications such as mail, calendar, contacts, and buddy lists that reside on the handset and interoperate with their PC counterparts. Integrating the mobile experience with the PC experience would enable consumers to access these heavily used applications from any device, while providing carriers with a key way to drive usage of their 2.5G and 3G networks and increase ARPU.

Another example of integrating desktop and mobile clients would be the ability to send text messages from a desktop e-mail client to a cell phone. The current practice of sending Short Message System (SMS) text messages is limited mostly to teenagers because of the awkwardness of using the phone keypad for data entry. Adding the ability to

compose the message on a desktop e-mail client would increase use of SMS text messaging by adults, which in turn would result in greater airtime usage on the part of cell phone users receiving these messages..

Telecom companies are already taking advantage of .NET-connected software to extend the functionality of their existing products and generate new revenues. Consider the following examples.

Example 1: CNM Networks. CNM Networks, one of the largest providers of IP networking in the world, used .NET-connected software to create a Voice Assistant service that provides on-demand conference calling from any device, integrated into Outlook. The system requires no bridges, special phone numbers, or sign-in codes, and users can select any number of people to participate in a conference. In just two months, CNM created over 30 voice conferencing features as Web services. The solution significantly reduces the cost of providing conference services—by 50 percent for large enterprises; 60 percent for small to medium companies ; and 70 percent for small offices and home offices.

Example 2: British Telecom. British Telecom, the major telephone service provider in Great Britain, added new functionality to its conference calling system with a .NET-connected solution. The new functionality enables anytime, any-device voice conferencing integrated with Microsoft Outlook and Windows Messenger. Using Visual Studio.NET and .NET Framework, BT worked with Microsoft to integrate conference call setup and management directly into Outlook and Windows Messenger. Now customers can use Outlook to select contacts and schedule a time. The BT solution sends the Outlook data to its conference bridge, schedules the time of the conference call, and assigns the numbers to call. Participants can join from PCs, mobile devices, and standard handsets. By simplifying the process of making conference calls and extending this capability to additional devices, this solution will result in added usage of BT's conference calling system, increasing the company's revenues.

Add Totally New Products

With .NET-connected software, telecom companies can easily and cost-effectively roll out new products, generating highly profitable new revenue streams. For example, companies could create a .NET-connected application that would integrate the .NET Alert service with Microsoft Exchange Calendar Service and the MapPoint Web service to notify users of upcoming appointments and provide them with directions to get there. Such an application would check the user's calendar for the next appointment time and location; determine the user's current location; get directions, map, and travel time; and send the user an SMS or Multimedia Messaging System (MMS) alert—complete with map—at the appropriate interval before the appointment.

.NET-connected software could also create and manage customer-centric portals, providing information unique to the needs of each customer and making that information accessible from a wide range of devices. A particular customer might choose, for instance, to access all of her bank accounts from a single page, and to have that

information delivered securely to her Smartphone or Blackberry. Such customer-centric portals offer added value to customers, helping companies increase their retention of existing customers as well as adding new ones.

And, by using .NET-connected software to expose core network capabilities on a wholesale basis to other developers and partners, companies could create an entirely new revenue stream based on transaction fees. For instance, by exposing its mobile and billing platforms, a company could allow third parties to bill mobile e-commerce transactions to customers' phone bills. As an example, a gaming developer could incorporate your billing Web service into a game that can be downloaded onto mobile devices. When users download the game, its cost would be billed to their account with you—and you would collect a piece of the revenue via a transaction fee.

Example: Telekom Austria. Telekom Austria developed a media-on-demand application using Visual Studio .NET. The application, which runs on Windows 2000 Server and uses Windows Media technology (both .NET-connected platforms) offers subscribers the ability to access media on demand—for example, downloading videos. This new service is motivating Telekom Austria customers to subscribe to its asymmetric digital subscriber line (ADSL) service, increasing both ARPU and data traffic.

.NET: The Easiest, Most Cost-Effective Way to Implement Web Services

You've seen examples of how .NET-connected software can reduce your operating costs, help you add new capabilities to existing products to increase revenues, and enable you to roll out brand-new products that extend your business into new areas. But why choose .NET over other Web services approaches? The answer is that only .NET-based connectivity across the Microsoft platform offers:

- **A complete, end-to-end solution for building, hosting, deploying, and using Web services.** The Microsoft Windows Server System, built from the outset with Web-based connectivity in mind, offers the most complete platform for building and deploying connected solutions. Microsoft's .NET-enabled servers, services, clients, and tools can help telecom companies easily and cost-effectively create solutions that connect all their systems—from back-end office systems to Windows-based desktop systems to the latest generation of mobile devices, such as Tablet PCs, Blackberries, and Smartphones.

- **Easy-to-use tools and services that reduce development costs and get solutions to market faster.** Powerful tools like the Windows .NET Framework and Microsoft Visual Studio .NET—the only development environment built from the ground up for Web services—make it possible to create value-added applications swiftly and inexpensively, so organizations can start realizing their benefits sooner. For example, based on the new middleware J2EE reference application with EJBs and an equivalent .NET reference application with C# components, optimizing and configuring a .NET application takes two weeks of labor, compared to ten weeks for J2EE. Further contributing to the high degree of developer productivity is Microsoft's library of "plug and play" data services, such as .NET Passport (for authentication), .NET Alerts (for notification),

MapPoint (for geographic-based services, such as maps and directions), and .NET My Services (for functions such as calendar, inbox, and so forth). These services enable developers to cost-effectively add new value to solutions without reinventing the wheel.

Example 1: Verizon Communications. Verizon has 300,000 voice lines and 3.5 million data circuits in lower Manhattan. After the September 11 terrorist attacks, which severely damaged the lower Manhattan site and shut down millions of circuits, Verizon needed to assess the impact of the outage as rapidly as possible. Using Visual Studio.NET, Verizon was able to re-engineer a data tracking and analysis tool and put it into production in *just six days*. The new application is managing a bigger circuit load than before, monitoring half a million circuits per hour and handling 8 GB of data per minute—a 400 percent increase over the previous application's capacity.

Example 2: T-Mobile. To drive new revenue streams, T-Mobile decided to build a service platform capable of connecting mobile users with business applications running on any platform and in any location. With this solution, mobile users can easily access their messaging, ERP, or CRM systems from a broad range of mobile devices. By choosing the Microsoft platform and .NET technologies, T-Mobile was able to create a scalable and reliable service platform and launch the service in less than a year, minimizing development and deployment costs while accelerating time to market.

Example 3: Scandinavian Airlines. Scandinavian Airlines used Visual Studio .NET and the Microsoft Mobile Internet Toolkit to create an application that provides real-time access to flight status and the ability to rebook flights, delivering the information to any device (including cell phones, pagers, and PDAs), any time, anywhere. With Microsoft's tools, the company was able to implement a mobile interface to its existing applications *in one week*. "Visual Studio .NET, along with the Microsoft Mobile Internet Toolkit…allowed us to deliver customized pages for myriad devices quickly and cost-effectively, plus we have the flexibility to build in exciting new features that will help SAS serve customers even better in the future," says Peter Müller, deputy director of Scandinavian's IT Group.

- **A significantly lower total cost of ownership.** NET-connected solutions have been proven to have a significantly lower TCO than other Web services solutions. This is partly due to the fact that they run on Intel-based hardware, which costs much less than other hardware platforms; and partly due to the ease of working with the .NET development tools, which dramatically increase developer productivity.

  Example 1: Verizon Communications. Verizon moved its call center and customer billing system off mainframes and onto the Microsoft platform, including SQL Server, Windows, and the .NET Framework. The solution, which manages over 9 TB of data and services more than 7,500 concurrent support representatives, is saving the company $10 million per year in hardware costs

alone. And the time it takes to make a functional or process change is one-fifth to one-tenth what it had been in the mainframe environment.

Example 2: Broadwing. The Broadwing solution mentioned earlier, which used .NET technology to automate the DSL order management and provisioning process, resulted in a 300 percent improvement in employee productivity, significantly increased ROI, and made the DSL segment of the business much more profitable. It also resulted in much higher customer satisfaction.

Moving to .NET: Six Steps You Can Take

Everything a telecom company needs to build, host, deploy, and use Web services is currently available from Microsoft, including:

- Scalable, reliable platforms for running .NET-connected applications, such as Microsoft Windows XP and Windows CE
- A robust, powerful Windows Server System, optimized for securely and reliably deploying, managing, and operating .NET-connected solutions
- Plug-and-play data services for cost-effectively adding value to .NET-enabled applications
- Smart clients that consume Web services natively
- A rich set of developer tools that enhance developer productivity and speed time to market

Together, these add up to a flexible, cost-effective platform for quickly integrating and extending existing telecom applications and products and building new ones—which, in turn, will help telecom companies to reduce costs, increase revenues, and decrease response time. That's why Gartner positions Microsoft as a leader in the move toward Web services, giving it the highest ranking for both completeness of vision and ability to execute.[1]

Here are six steps your company can take today to get .NET-connected:

1. Learn more about .NET and the next-generation Internet by visiting www.microsoft.com/net.
2. Discuss your organization's Web services strategy with your IT department.
3. Meet with a consultant who specializes in .NET-connected software—such as Accenture or Hewlett Packard—and explore the impact it can have on your organization.

---

[1] Gartner Research, October 2002.

4. Take the first step towards connecting your systems more closely—both internally and with those of external partners—by deploying pilot projects using Web services based on .NET-connected software.

5. Use vendors and partners that have a roadmap for making their applications accessible as standards-based Web services.

6. To ensure interoperability and long-term value, ask your IT department to base all future solutions on .NET-connected software.

Microsoft .NET is the solution telecom companies have been looking for to help them successfully deal with the challenges facing the industry today. With .NET-connected software, you'll be able to cost-effectively integrate your systems to reduce OSS/BSS and achieve new operational efficiencies. You'll also be able to enhance your existing products to generate more revenue and roll out totally new products that can extend your business into new areas and drive new revenue streams. With more than 35,000 certified partners and a full range of .NET-connected technologies, Microsoft and Microsoft partners are uniquely positioned to help you take advantage of all the benefits that Web services have to offer.