

# Vernetzte Mess- und Automatisierungstechnik

## Windows XP und .NET – zwei neue Technologien für Entwickler und Automatisierer

**O**bwohl Anfang der 90er Jahre die landläufige Meinung vorherrschte, dass Windows 3.x als Betriebssystem für Mess- und Automatisierungsanwendungen kaum akzeptabel sei, so ist doch bemerkenswert, in welch vielfältigen Bereichen sich Windows in der Zwischenzeit durchgesetzt hat – auch die hartnäckigsten Windows-Gegner sind mittlerweile umgeschwenkt. Auch wenn aus technischer Sicht für die Mess- und Automatisierungstechnik andere Betriebssysteme geeigneter erscheinen, ist es zur Zeit auf Grund des umfangreichen Angebots von Entwicklungsumgebungen und unterstützter Hardware sowie des weit verbreiteten Ingenieurswissens für diese Plattform einfach ökonomischer, auf Windows-basierte Lösungen zu setzen. Mit Windows XP und .NET werden auch die Mess- und Automatisierungstechniker wieder mit neuen Technologien konfrontiert. Der Beitrag erläutert konkret die Anwendbarkeit von Windows XP und .NET in der Mess- und Automatisierungstechnik.

Nun kommt Windows XP, das beste Betriebssystem aller Zeiten, das alle zufrieden stellen soll, so Microsoft. Fakt ist: Mit Windows XP wurde die schon vor geraumer Zeit in Aussicht gestellte Vereinigung von Windows/DOS und Windows NT Realität. Die auf MS DOS basierende Produktlinie, die mit Windows 98 ihren Höhepunkt und mit Windows 98 SE und Windows ME noch zwei Erweiterungen erfahren hatte, wurde eingestellt. Es gibt im Mainstream-Bereich nur noch ein Windows (mit zwei Geschmacksrichtungen: Home und Professional), das sich von Windows NT ableitet, aber mit allen Vorzügen von Windows ME und Windows 2000 ausgestattet wurde.

### Autoren

Dipl.-Ing RAHMAN JAMAL und Dipl.-Ing. (FH) JOCHEN KLIER sind leitende Mitarbeiter der National Instruments Germany GmbH; Konrad-Celtis-Straße 79, D-81369 München  
Fon: 089/741313-0, Fax: 089/7146035  
eMail: info.germany@ni.com

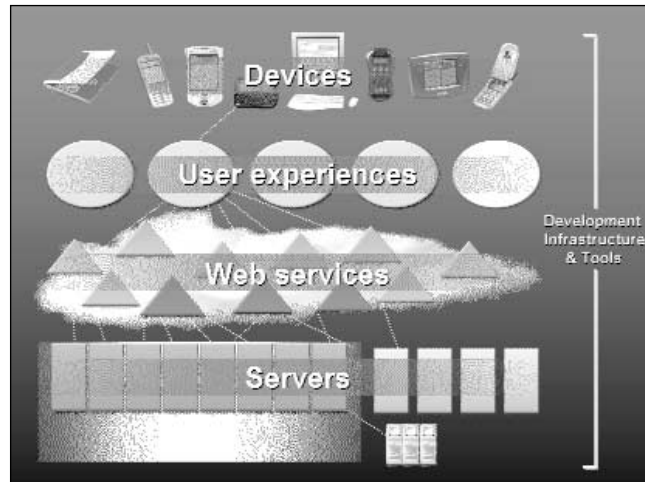


Abb. 1:  
Das .NET Framework [1]

### Windows XP – kein müdes Update

Windows XP stellt einen neuen Meilenstein in der Evolution von Windows dar – nicht nur auf der Oberfläche, sondern auch unter der Haube. Es ist deutlich robuster und sicherer im Vergleich zu seinen Vorgängerversionen, besitzt ein extrem leistungsfähiges Dateisystem, unterstützt Multimedia in allen Variationen und vor allem stellt es eine ganze Reihe von Netzwerk-Features zur Verfügung. Alleine diese Aufzählung macht deutlich, dass Windows XP nicht ein weiteres ‚müdes‘ Update ist. Gerade Betriebssicherheit und zusätzliche netzwerkorientierte Möglichkeiten wie ‚Remote Desktop‘, ‚Remote Assistance‘ oder ‚Application Sharing‘ mussten in der Vergangenheit mit einem erheblichen Mehraufwand an Engineeringkosten in Fern-Mess- und -Steuerungsapplikationen integriert werden. Hier sorgt Windows XP für eine solide Basis.

### .NET als Grundlage

Der wohl interessanteste, wenn auch derzeit kontroverseste Aspekt von Windows XP ist sicherlich die damit einhergehende Einführung der .NET-Technologie. Das Ziel dieser Microsoft-Initiative ist es diesmal nicht, wie bei MSN und anderen früheren proprietären Ansätzen, ein eigenes Internet ins Leben zu rufen, sondern vielmehr die bestehenden Netzwerktechnologien und -Infrastrukturen

einzusetzen, um Informationen universell und plattformübergreifend nutzbar zu machen. XML-basierte Web-Dienste sollen dabei die zentrale Rolle spielen. Solche Web-Dienste könnten zum Beispiel Staudaten an Routenplaner übermitteln, global verfügbare persönliche Terminkalender verwalten, Informationen über Kino- und Theateraufführungen bereitstellen und auch gleich noch das bargeldlose Bezahlen erledigen. Das gibt es alles schon? Richtig. Der Unterschied liegt aber darin, dass zur Zeit alle existierenden Lösungen auf uneinheitlichen Technologien fußen und somit eine universelle Einsetzbarkeit nicht gewährleistet ist. Unter „universeller Einsetzbarkeit“ ist hier zum Einen die Möglichkeit gemeint, dass unterschiedliche Applikationen wie Fahrzeug-Navigationssystem, Terminkalender und E-Mail-Client über das standardisierte Datenformat XML miteinander kommunizieren. Z. B. könnten für einen Außendienstmitarbeiter bei einem Stau, über den das Navigationssystem von einem Web-Dienst informiert wurde, automatisch Anpassungen im Kalender vorgenommen und alle Betroffenen per E-Mail über die Terminverschiebung benachrichtigt werden. Zum Anderen bedeutet ‚universelle Einsetzbarkeit‘ aber auch, dass nicht nur PC-basierte Windows-Systeme auf die Daten der Web-Dienste zugreifen können, sondern auch andere ‚Smart Devices‘ wie Mobiltelefone oder PDAs.

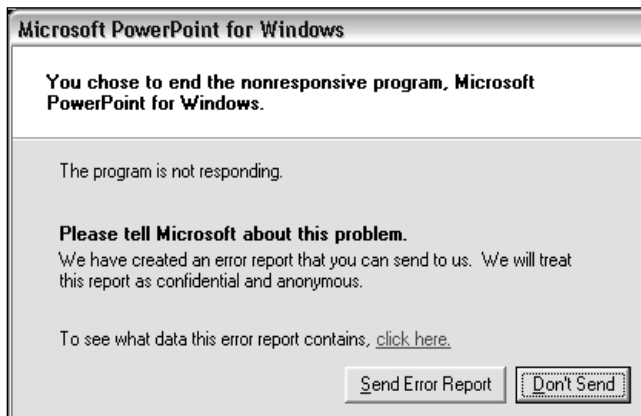
Interessant für den Mess- und Automatisierungstechniksektor wird die .NET-Technologie dann, wenn man sie sich abstrahiert als Möglichkeit zur internetbasierten Kommuni-

kation vorstellt, die einem offengelegten Standard folgt. Dann braucht man nicht mehr viel Fantasie, um sich die einfache Realisierung von Applikationen z. B. für das Betrachten von Prozessdaten einer Automatisierungseinrichtung auf einem PDA oder beliebigen anderen Plattformen vorstellen zu können.

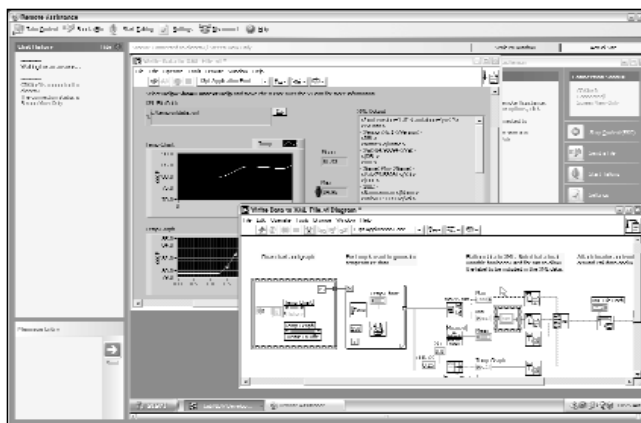
Voraussetzung für den Erfolg von .NET ist aber die von Microsoft angekündigte vollständige Offenlegung der Technologie. Es bleibt abzuwarten, ob die Redmonder diesmal wirklich mit offenen Karten spielen oder ihrer altbekannten Taktik folgend durch unvollständige Dokumentation oder patentrechtliche Stolperfallen ihre Monopolstellung weiter auszubauen suchen. Nicht aus dem Blickwinkel verlieren sollte man bei all dem Öffentlichkeitsrummel um .NET die auf Java basierende ‚SunONE-Initiative‘ von Sun, die ähnliche Ziele verfolgt. Sun ist es bereits gelungen, einige weitere Hersteller in ihr Boot zu ziehen. Sofern das Ziel beider Firmen tatsächlich sein sollte, leistungsfähige und offene Lösungen für die Anwender zu realisieren, wären sie wohl gut beraten, anstatt immense Anstrengungen in konkurrierende Systeme zu investieren, sich an einen Tisch zu setzen und damit Synergie-Effekte zu nutzen. Dass so eine Zusammenarbeit durchaus Früchte tragen kann, sieht man beispielsweise bei den endlosen Diskussionen um den ultimativen Feldbus. Zwar wurde dabei keine Einigkeit auf Hardwareebene erzielt, immerhin aber eine für den Anwender sinnvolle Kommunikationslösung über den OPC-Standard bzw. das darauf aufbauende OPC DX. OPC DX entstand unter der Mitwirkung von Feldbus-Nutzerorganisationen wie Interbus-S, FF, ODVA und PNO mit dem Ziel, einen durchgängigen Datenaustausch auf der Systemebene zu ermöglichen.

## Windows XP in der Praxis

Auffällig ist die nach der Installation durchzuführende Aktivierung des Betriebssystems, ohne die es nach 30 Tagen seinen Dienst verweigert. Zu diesem Zweck wird aus der Rechnerhardware eine eindeutige ID erstellt, die entweder über das Internet oder per Telefon an Microsoft übermittelt wird, worauf man eine Aktivierungs-ID erhält, die das Betriebssystem für die uneingeschränkte Nutzung freischaltet. In die Rechner-ID gehen Komponenten wie z. B. die MAC-Adresse der Ethernet-Karte, die Seriennummer der Festplatte(n) und der Prozessortyp ein, nicht aber Komponenten wie GPIB-Schnittstellen oder Datenerfassungskarten. Ändern sich mehr als drei der berücksichtigten Komponenten, muss die Aktivierung erneut durchgeführt werden. Da dieser Vorgang bei Aktivierung über das Inter-



**Abb. 2:**  
Wer hier auf ‚Send‘ klickt, schickt unter Umständen sensible Daten an Microsoft



**Abb. 3:**  
Mit Remote Assistance können Entwickler sich bei ihrer Arbeit gegenseitig unterstützen

net aber nur einige wenige Sekunden in Anspruch nimmt und zu keiner Zeit persönliche Daten abgefragt oder übertragen werden, erscheint die Aufregung darüber eher etwas polemisches.

Viel eher angebracht ist Skepsis gegenüber den neuen Features, die das Potenzial bergen, gewollt oder ungewollt persönliche oder sensible Daten an Microsoft zu übermitteln. Das ist zum einen die neue Option, die den Anwender vor die Wahl stellt, nach der unplanmäßigen Beendigung eines Programms einen Fehlerbericht an Microsoft zu schicken. Klickt der Benutzer hier auf ‚Senden‘, besteht die Möglichkeit, dass zuletzt von dieser Applikation geladene Daten mit in den Fehlerbericht gelangen.

Zum anderen ist dies die immer wiederkehrende Aufforderung, sich einen .NET Passport zu erstellen. Über den Passport erhält der Anwender eine eindeutig zuordenbare Identität, die für die Nutzung zukünftiger .NET-Web-Dienste notwendig sein wird und ebenfalls für die Nutzung von Windows-XP-Features wie den ‚Microsoft Messenger‘ vorausgesetzt wird.

## Erweiterte Netzwerkooptionen

Der Windows Messenger ist Teil eines Konzepts zur erweiterten Netzwerknutzung und ist vergleichbar mit Applikationen wie dem

‚AOL Instant Messenger‘ oder ‚Sametime‘ von IBMs Lotus Notes. Neben der Möglichkeit, anderen Teilnehmern in ‚Echtzeit‘ Nachrichten zukommen zu lassen, stellt der Microsoft Messenger gleichzeitig den Einsprungpunkt zu Applikationen wie ‚Remote Assistance‘ oder ‚Remote Application‘ dar.

Bei der Remote Assistance kann ein Anwender von einem anderen Hilfe anfordern und ihm Einblick auf seine Bildschirmoberfläche gewähren. Bei Bedarf kann er die Kontrolle über Maus und Tastatur auch an den Helfer abgeben und bei Vorhandensein einer Soundkarte samt Mikrofon und Lautsprecher sowie ausreichender Netzwerkbandbreite sich sogar mit seinem Gegenüber unterhalten. Interessant ist Remote Assistance zum Beispiel für räumlich verteilt arbeitende Entwicklerteams, die sich so bei ihrer Arbeit gegenseitig unterstützen können. Da sich der Messenger wegen der Anmeldung über den .NET Passport aber nur bei vorhandener Internet-Anbindung starten lässt, kann Remote Assistance auch über das Verschicken einer E-Mail oder über eine freigegebene Datei initiiert werden. Remote Application ist im Prinzip eine Variante von Remote Assistance, bei der aber nicht der gesamte Desktop sondern nur eine einzige Applikation zur gemeinsamen Benutzung freigegeben wird. Im Unterschied zu Remote Assistance kann Remote Application aber nur unter Verwendung des Messengers genutzt werden.

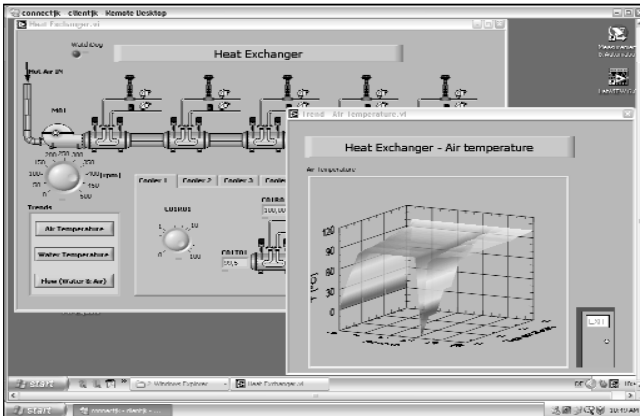


Abb. 4: Fernwartung mit Remote Desktop

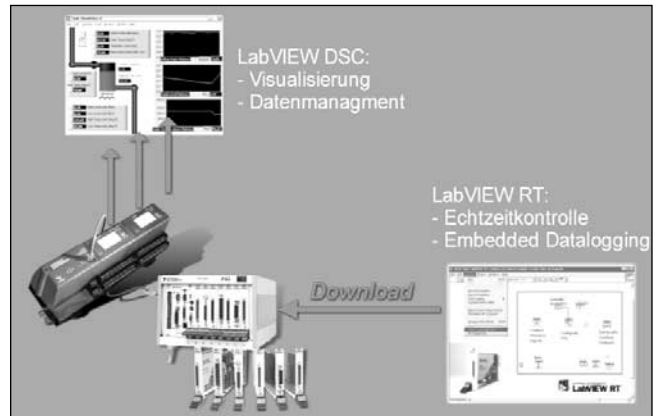


Abb. 5: Echtzeitsysteme von National Instruments werden mit LabVIEW RT programmiert. LabVIEW DSC kann als System zum Bedienen und Beobachten oben aufgesetzt sein

Fernwartungsaufgaben lassen sich mit Windows XP leicht mit der integrierten Funktion ‚Remote Desktop‘ realisieren. Remote Desktop basiert auf der bekannten Windows-Terminal-Server-Technologie und ermöglicht es, sich auf einem über Intra- oder Internet vernetzten Rechner einzuloggen und ihn genauso zu bedienen, als wäre man vor Ort. Laufende Applikationen müssen dabei nicht beendet werden.

**Stabilität, Performance, Echtzeit**

Viele Detailverbesserungen, aber auch neu überdachte Basisfunktionalitäten machen Windows XP deutlich stabiler und leistungsfähiger als die Vorgängerversionen. Für die erhöhte Stabilität zeichnen vor allem die Funktionen ‚System Restore‘ und ‚Driver Rollback‘ verantwortlich, mit denen das System z. B. nach allzu mutigen Beta-Installationen wieder in einen funktionierenden Zustand versetzt werden kann. Das neue Konzept der ‚Side-by-side-DLLs‘ adressiert das leidige Problem inkompatibler DLL-Versionen. Bisher litt die Zuverlässigkeit eines PCs häufig darunter, dass neu installierte Software DLLs, die auch von anderen Applikationen genutzt wurden, durch neuere Versionen ersetzt. Kleine Inkompatibilitäten der verschiedenen DLL-Versionen

zogen so häufig große Probleme nach sich. Side-by-side-DLLs werden in speziellen Systemverzeichnissen abgelegt und wie der Name schon sagt, können mehrere Versionen friedlich nebeneinander koexistieren. Applikationen, die auf diesen DLL-Typ zugreifen, können nun explizit angeben, mit welcher Version sie arbeiten möchten. Auch das Starten von Applikationen wird durch ein intelligentes ‚Prefetching‘ merklich beschleunigt.

Wie sieht es nun mit der ‚Echtzeit‘ aus? Spätestens nach diesen Ausführungen dürfte jedem klar werden, dass auch Windows XP nicht für die Gruppe der harten Echtzeitanwendungen ausgelegt ist. Somit bleibt das alte Problem Echtzeit in allen Bereichen der ‚MSR-Technik‘ nach wie vor aktuell. Mit der Durchsetzung von Windows wuchs aber auch die Anzahl der Lösungsansätze, die sich mit den Problemen von Echtzeit und Determinismus in unterschiedlichsten Ausprägungen auseinandersetzen. Konzentrierten sich die ersten Lösungen auf einfache Entkopplung der MSR-Anwendung über Datenpuffer in Form von FIFOs auf der Erfassungshardware, so wird heute durch das Auslagern von Softwarekomponenten auf die Hardware eines intelligenten Subsystems Windows umgangen. Solche Subsysteme können z. B. als PCs im Steckkartenformat oder als externe modulare Lösungen ausgelegt sein oder aber auch ein komplettes PXI-System umfassen.

**Ausblick**

Wie immer sind mit dem Erscheinen einer Windows-Version bereits weitere Nachfolger mit spektakulären Codenamen im Gespräch. Das für das Jahr 2005 anvisierte ‚Blackcomb‘ soll die nächste Generation von Windows einläuten. Vor Blackcomb soll es im Jahr 2003 erst noch einmal einen Zwischenschritt geben, der bereits den Codenamen ‚Longhorn‘ besitzt. Natürlich dringen solche Neuigkeiten nicht zufällig nach außen, sondern es handelt sich dabei um gezielte Indiskretionen, die das Interesse an der nächsten Generation stetig steigern sollen. **TEST**

**Literatur**

- [1] R. Jamal, J. Klier: Was bringen XP und .NET?, Elektronik Heft 8, 2002.
- [2] Peter Siering: Das Microsoft-Internet, c't Heft 4, 2002.
- [3] Rahman Jamal, Andre Hagedstedt: LabVIEW – Das Grundlagenbuch, Addison-Wesley, 2001.
- [4] Rahman Jamal, Hans Jaschinski: Virtuelle Instrumente in der Praxis (VIP 2002) Hüthig-Verlag, 2002.

www.publish-industry.net  
more @ click TK3C0302

**LESERTIPP**

*Sie haben Anregungen oder Verbesserungsvorschläge für die Nächstjahresausgabe des TEST KOMPENDIUMS?*

**Wir freuen uns auf Ihre Ideen.  
Rufen Sie uns an: +49/89/500 383-0**

**publish industry**  
TECHNIK KOMMUNIZIEREN

Gollierstraße 23 · D-80339 München · Fon. +49/89/500383-0 · Fax. +49/89/500383-10 · info@publish-industry.net · www.publish-industry.net