

Oliver Baumann

## Die neue „OPC-Freiheit“

Bisherige, eng an die Microsoft-Technologie angelehnte Standards für den Datenaustausch zwischen Applikationen bedingen zum Teil proprietäre Lösungen und führen bei web-basierten Systemen oft zu Konflikten mit Firewalls. Abhilfe schaffte die Einführung des OPC-XML-DA-Standards auf der Grundlage der OPC-Foundation-Spezifikation, der nun auch den Einsatz von Java-Architekturen für die Prozessdaten-Visualisierung erschließt.

Die von der OPC-Foundation erstellten und herausgegebenen OPC-Standards gehören heute zu den wichtigsten Standards für die hersteller-unabhängige Kommunikation in der Automatisierungstechnik. Die erste „OPC Specification Version 1.0“ wurde im August 1996 herausgegeben, ihr folgten im Laufe der Jahre weitere Standards für diverse Teilbereiche der Kommunikation. Allen OPC-Spezifikationen ist gemein, dass sie auf den COM/DCOM-Technologien von Microsoft basieren. Durch diese Abhängigkeit konnten OPC-konforme Systeme nur aufwendig und nicht standardisiert plattform-unabhängig genutzt werden.

Erst die im Jahr 2003 veröffentlichte „OPC XMLDA 1.00 Specification“ und die überarbeitete und im Dezember 2004 herausgegebene Version „OPC XMLDA

1.01 Specification“ basieren nicht mehr auf COM/DCOM, sondern auf hersteller-unabhängigen Technologien wie XML (Extensible Markup Language), Web-Services und SOAP (Simple Object Access Protocol). Damit ist aber erstmals „echte“ Interoperabilität und Plattform-Unabhängigkeit möglich.

Ein OPC-Server, der die „OPC XMLDA 1.01 Specification“ einhält, bietet einen Web-Service an, welcher über HTTP aufgerufen werden kann. Über diesen Web-Service kann auf Funktionen wie „Subscribe“, „Read and Write“, „Status“ und „Browse“ eines oder mehrerer OPC-DA-Server zugegriffen werden. Was bringt das in der Praxis?

Web-Services sind heute der De-facto-Standard zur offenen, plattform- und sprach-unabhängigen Kommunikation

zwischen IT-Systemen. Die großen Softwarehersteller wie SUN, IBM, Microsoft oder SAP integrieren und nutzen Web-Services in ihren Produkten. Der Vorteil liegt auf der Hand: Durch einen hohen Grad an Interoperabilität tauchen beim Integrationsprozess deutlich weniger Probleme auf.

### Der Nutzen von Web-Services

Bei der Nutzung von Web-Services zur Kommunikation stellt der Anbieter eines Web-Service eine standardisierte Schnittstelle zur Verfügung, die die Interoperabilität gewährleistet. Ein Client als Nutzer des Web-Service greift auf diese Schnittstelle zu. Dabei spielt es keine Rolle, mit welcher Technik der Client diesen Zugriff ausführt. So kann die Anwendungslogik hinter dem Web-Service die .net-Technologie nutzen, während die Anwendungslo-

(Grafik: Computer & AUTOMATION)

OPC-XML-DA ermöglicht eine Architektur, die J2EE und .net verbindet. Dazu werden standardisierte Techniken wie SOAP und Web-Services genutzt.

gik hinter dem Client komplett in J2EE (Java 2 Platform, Enterprise Edition) realisiert ist.

Der Datenaustausch zwischen dem Web-Service-Anbieter und dem Client erfolgt mit Hilfe von SOAP-Nachrichten. SOAP stützt sich auf die Dienste anderer Standards wie etwa XML. Zur Übertragung der SOAP-Nachrichten und damit zur Kommunikation mit Web-Services wird häufig HTTP (Hypertext Transfer Protocol) genutzt. Dies hat bei web-basierten Lösungen den Vorteil, dass es – anders als bei der Nutzung von DCOM – zu keinen Problemen mit vorhandenen Firewalls kommt. Somit ist endlich der Sprung zu einer wirklich offenen Architektur gelungen, die es ermöglicht, J2EE auch für die Prozessdaten-Visualisierung zu nutzen.

## Java zur Visualisierung

J2EE ist die Spezifikation einer Standardarchitektur, die das Zusammenspiel von verschiedenen Java-Standards regelt, um damit mehrschichtige, web-basierte Systeme zu entwickeln. Mittlerweile gibt es eine Vielzahl von J2EE-basierten Application-Servern, die diese Spezifikation umgesetzt haben. Hersteller dieser Application-Server sind Unternehmen wie SUN, SAP, Oracle, IBM oder BEA. J2EE-konforme Systeme bieten folgende Vorteile:

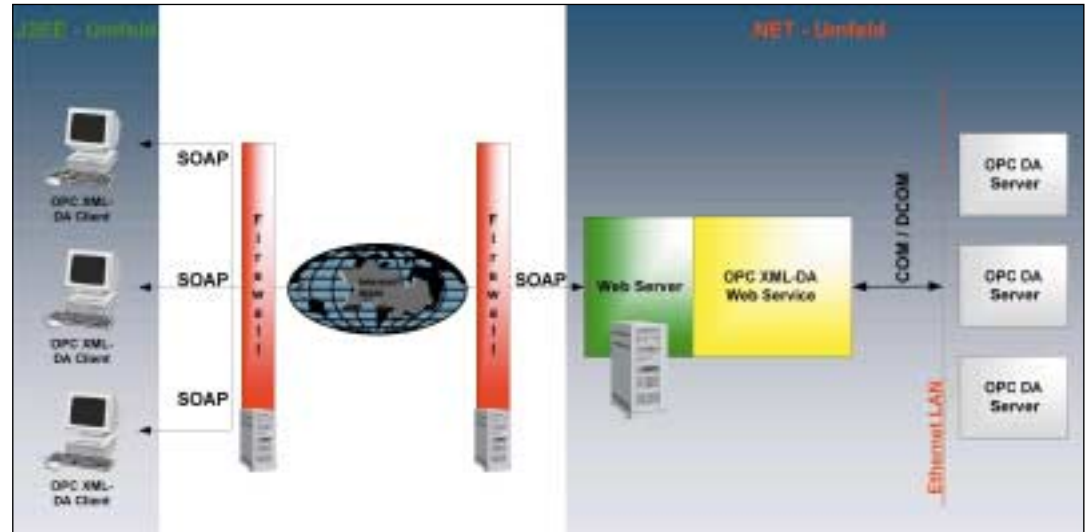
- ▷ ausgereifte Technologie, die sich durch Standardisierung, hohe Stabilität und hohe Verfügbarkeit auszeichnet;
- ▷ hohe Skalierbarkeit, wodurch ein J2EE-konformes System leichter an eine gesteigerte Last anzupassen ist;
- ▷ modularer Aufbau als gute Grundlage für mögliche Erweiterungen;
- ▷ ideale Integrationsmöglichkeiten für heterogene Systemumgebungen;
- ▷ Interoperabilität und Plattform-Unabhängigkeit und damit höheren Investitionsschutz bei eventuellen Änderungen der Systemumgebung.

Für die Nutzung des J2EE-Standards spricht weiterhin der Umstand, dass viele Portale auf diesem Standard basieren. Portale dienen unter anderem dazu, den Informationszugriff innerhalb eines Unternehmens effizienter zu gestalten. Somit lassen sich durch den Einsatz von OPC-XML-DA und J2EE auch die Informationen aus der Automation über Prozessdaten-Visualisierung in Portale integrieren.

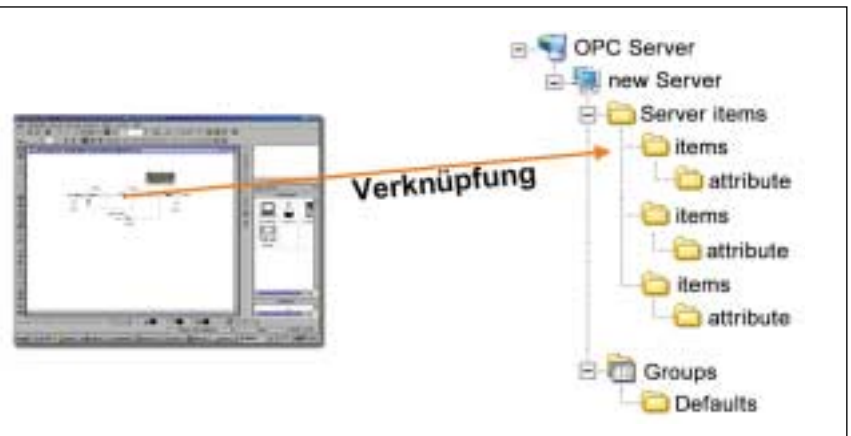
Auch für die OPC-Foundation geht der Trend immer weiter in Richtung XML und Web-Services und somit auch in Richtung Interoperabilität und Plattform-Unabhängigkeit. Dies zeigt sich beispielsweise daran, dass die OPC-Foundation momentan an der auf XML- und Web-Services basierenden OPC-UA (Unified Architecture) arbeitet. OPC-UA soll die Möglichkeit bieten, Systeme für OPC zu entwickeln, die Informationen über Plattformen von verschiedenen Herstellern über das Internet austauschen können.

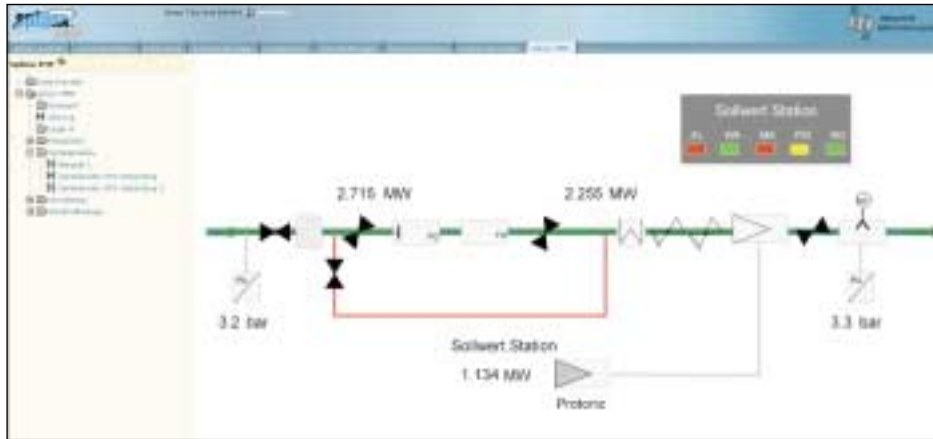
## J2EE in der Praxis

Ein Beispiel für ein Unternehmen, das auf OPC-XML-DA beziehungsweise J2EE zur Prozessdaten-Visualisierung sowie zur Automationssteuerung setzt, ist die *Mainova AG*. Dabei handelt es sich um einen Energie-Erzeuger, der den Großraum Frankfurt mit Strom, Gas und Wasser versorgt. Bisher setzte Mainova zur Visualisierung und Steuerung des Gasverteilnetzes ein Leitstandsystem ein, das proprietäre Bausteine für die Übermittlung



Ein Objekt wird bei der Grafik-Erstellung über Adressierung mit einem Server-Item verknüpft. Über diese Verknüpfung werden im Visualisierungs-Client die Daten über den Web-Service vom OPC-Server abgerufen und visualisiert.





Eine Grafik wird im SVG-Format (Scalable Vector Graphic) in ein Portal eingebunden und visualisiert über dynamische Objekte die OPC-Daten.

Um die Daten zu visualisieren und Steuerungen durchzuführen, wurde für die Mainova AG ein auf der „sphinx open JAVA Edition“ basierender Visualisierungs-Client erstellt. – Ein Ansatz, der in Summe eine schnelle und reibungslose Integration der OPC-XML-DA-Lösung in die auf Linux basierende IT-Architektur gewährleistet hat.

gh

von Daten und deren Visualisierung nutzte. Im Zuge einer Modernisierung wurden die Kommunikationsprotokolle auf den offenen OPC-Standard umgestellt.

Um die Kosten für Lizenzen und die Hardware so gering wie möglich zu halten, sollte eine Linux-basierte Lösung zum Einsatz kommen. Auch sollte für das System eine ausgereifte Technologie verwendet werden und es sollte bei Veränderungen in der IT-Struktur leicht portierbar

sein. Damit kam nur ein System auf Basis von OPC XML-DA und J2EE in Frage.

Die neu geschaffene Lösung setzt sich aus den zwei Hauptfunktionen Erstellung und Versorgung von Grundgrafiken und der Visualisierung und Steuerung innerhalb des Verteilnetzes zusammen. Zur Erstellung der Grundgrafiken und zur Verknüpfung der einzelnen Objekte mit OPC-Datenpunkten kam der „sphinx EMP Editor“ der *in-GmbH* zum Einsatz.

Nähere Informationen:  
oliver.baumann@in-gmbh.de



**Oliver  
Baumann**

ist Engineer IT-Solutions  
bei der in-GmbH,  
Konstanz.