



XML

Harald Schöning, Software AG

XML konnte im Februar diesen Jahres seinen fünften „Geburtstag“ feiern: Am 10. Februar 1998 wurde XML 1.0 zur „Recommendation“ beim World Wide Web Consortium (W3C). XML ist damit zwar kein de-jure-Standard, da das W3C kein offizielles Standardisierungsgremium ist, aber niemand kann bezweifeln, dass XML zum branchenübergreifenden de-facto-Standard geworden ist, der erheblichen Einfluss auf viele Gebiete der Informationstechnologie ausübt.

Als ich gebeten wurde, ein Themenheft „XML“ zusammenzustellen, war mir klar, dass man in einem einzigen Heft keinen annähernd umfassenden Überblick über die vielen Facetten von XML geben kann. Allein zum Thema „XML und Datenbanken“, das in diesem Heft mit einem kurzen Beitrag repräsentiert ist, erschienen im letzten Jahr schon drei deutschsprachige Bücher. Die Zahl der Bücher über XML-Anwendungen, Web Services, Programmierung mit XML usw. ist kaum noch zu überschauen. Selbst die Anzahl der XML-bezogenen Standards nimmt ständig zu, sowohl im „Kernbereich“ von XML (XML Schema, XPointer, XLink, XQuery usw.) als auch in dessen Anwendung (es seien nur die Web-Service-relevanten Standards erwähnt, wie SOAP, WSDL, UDDI usw.).

Im Jahre 1997 beschloss die Software AG, auf XML zu setzen und ein eigenes Datenbanksystem für dessen Speicherung zu entwickeln. So kam ich erstmals mit XML in Berührung. Damals wuss-

ten nur wenige Experten, was sich hinter diesem Kürzel verbirgt. Selbst an vielen Hochschulen war XML noch unbekannt. Inzwischen gibt es wohl kaum noch eine deutsche Universität, deren Informatiklehrstühle keine Forschung im Zusammenhang mit XML betreiben. Im *Electronic Business* ist XML nicht mehr wegzudenken, und sogar im *E-government* wird es verwendet.

Es gibt also eine Fülle von XML-bezogenen Themen, aus denen ich für dieses Heft eine Auswahl zu treffen hatte. Ich habe mich dafür entschieden, einen Bogen zu schlagen, der von einer Standardisierungsaktivität (XQuery) über die systemnahe Implementierung bis zu eher theoretischen Betrachtungen reicht, um schließlich eine Anwendung von XML (Web Services) zunächst generell und dann am konkreten Beispiel zu behandeln. Selbstverständlich ist diese Auswahl rein willkürlich. Ich hoffe aber dennoch, ein interessantes Spektrum von Themen gefunden zu haben.

XML ist eine Metasprache: Mit XML können Sprachen mit eigenen Vokabularen geschaffen werden, die auf das jeweilige Einsatzszenario zugeschnitten sind. So behalten Daten auch im XML-Format ihre Semantik. Diese Eigenschaft von XML ist die wesentliche Voraussetzung für eine Weiterverarbeitung (im Gegensatz zur reinen Präsentation, die bei HTML den Schwerpunkt bildet) und damit für den Datenaustausch zwischen Programmen. Auf der Anwendungsebene bedeutet das, dass HTML auf den Bereich Maschine-

Mensch (oder *Business to Customer*) abzielt, während XML auch den Bereich Maschine-Maschine (*Business to Business*) abdeckt.

Entsprechend dieser Orientierung gibt es auch standardisierte Methoden, einzelne XML-Dokumente weiterzuverarbeiten, wie z.B. XSLT-Stylesheets. Mengenorientierte Verarbeitung hingegen ist in einem Datenaustauschszenario, in dem jede Nachricht ein einzelnes XML-Dokument ist, noch nicht erforderlich. Aber schon aus dem reinen Datenaustausch ergibt sich auch die Notwendigkeit, XML-Dokumente aufzubewahren, z.B. um rechtlichen Anforderungen zu genügen. Fast zwangsläufig entstehen also Sammlungen von XML-Dokumenten. Darüber hinaus kann XML auch als eigentliches Speicherformat verwendet werden – warum sollte man Information in ein anderes Format überführen, wenn das vorliegende Format gut handhabbar ist? Zu dieser Handhabbarkeit gehört unter anderem eine mächtige Abfragefunktionalität, die es ermöglicht, Informationen wiederzufinden und zu extrahieren, und zwar auch bei großen Datenmengen.

Leider gibt es noch keine standardisierte Abfragesprache, die eine entsprechende Funktionalität über großen Mengen von Dokumenten anbietet. Als standardisierte Sprache, die eine Navigation in einem einzelnen XML-Dokument erlaubt, gibt es XPath, das z.B. in XSLT eingesetzt wird. XPath reicht jedoch nicht aus, um aus Mengen von



Dokumenten flexibel Anfrageergebnisse aufzubauen. Das W3C arbeitet aber schon seit 1999 an der Definition einer mächtigeren Sprache mit dem Namen XQuery als Nachfolger von XPath.

Der Beitrag von *J. Harbarth* befasst sich mit dieser Sprache. Dazu stellt die Autorin zunächst XPath vor und zeigt dessen Defizite auf. Anschließend wird gezeigt, welche Konzepte XQuery bietet, um diese Defizite zu beseitigen. Der Leser bekommt einen Eindruck von XQuery und seinem Umfeld im W3C.

Wie eine solche Sprache effizient implementiert werden kann, zeigt der Beitrag von *P. Fankhauser* und *P. Lehti*. Dabei wird besonders auf Aspekte der Typbehandlung Wert gelegt. Gerade bei großen Datenmengen ist eine ausgefeilte Optimierung von Anfragen unerlässlich. Einige Techniken hierzu werden in dem Beitrag aufgezeigt.

Um ein komplementäres Thema, nämlich die transaktionsorientierte und effiziente Speicherung von XML-Dokumenten, geht es im Beitrag von *S. Helmer*, *C.-C. Kanne* und *G. Moerkotte*. Die Autoren erarbeiten die Anforderungen an eine angemessene Speicherung von XML-Dokumenten. Dabei zeigt sich schnell, dass für die Speicherung von Daten im XML-Format genauso wie für Daten in herkömmlichem (relationalem) Format ein Datenbanksystem unerlässlich ist. Hierbei stellt sich die Frage, ob ein relatio-

nales Datenbanksystem auch für die Speicherung von XML verwendet werden kann. Die Autoren halten dies nicht für den optimalen Ansatz und präsentieren ein speziell für die Speicherung von XML entworfenes Datenbanksystem als Beispiel für XML-spezifische Systeme.

Nicht nur für die Speicherung, sondern auch für die Verarbeitung in Werkzeugen ist eine genaue Kenntnis der zu erwartenden XML-Dokumentstrukturen hilfreich. Solche Strukturen lassen sich mit Grammatiken beschreiben. *H. Lobin* geht auf Dokumentgrammatiken – wie DTD oder XML Schema – ein und untersucht ihre Eigenschaften und Anwendungsmöglichkeiten.

Web Services sind ein Vorzeigebeispiel für eine XML-basierte Maschine-Maschine-Kommunikation. Nicht nur zum Erteilen von Aufträgen an Web Services oder zum Versenden der Service-Antwort wird XML benutzt – eine ganze Infrastruktur aus Web-Service-Beschreibungen und Verzeichnisdiensten wurde basierend auf XML definiert. In seinem Beitrag hinterfragt *M. Jeckle* kritisch die Definition von Web Services und gibt einen Überblick über die Begrifflichkeiten, Standards und Techniken in diesem Bereich.

U. Coester stellt schließlich ein Beispiel für den Einsatz solcher Web Services in einem konkreten Anwendungsgebiet vor. Im Strom-

markt müssen seit der Deregulierung regelmäßig Kundendaten zwischen Energieversorgungsunternehmen ausgetauscht werden – ein bisher aufwändiger Prozess, der durch den Einsatz von Web Services und XML vereinfacht werden kann.

Mein Dank gilt den Autoren, die Beiträge zu diesem Themenheft geliefert haben, und den Hauptherausgebern für ihre Unterstützung. Den Lesern wünsche ich, dass die Beiträge Ihr Interesse wecken und einen guten Einblick in einige Aspekte von XML geben.

Harald Schöning



Dr. Harald Schöning hat an der Universität Kaiserlautern promoviert. Zurzeit arbeitet er als Architekt für den Tamino XML Server der Software AG.

Adresse: Software AG, Umlandstr. 12,
D-64297 Darmstadt,

E-Mail: Harald.Schoening@softwareag.com