



Orientation in Objects

Web Services und Open Source

) Schulung)

AUTOR



Thomas Bayer
Orientation in Objects GmbH

) Beratung)

Veröffentlicht am: 31.8.2004

WEB SERVICES UND OPEN SOURCE

) Entwicklung)

Marketing Schlagworte und kommerzielle Produkte beherrschen die Diskussion über Webdienste. Zeitschriften und Portale beschreiben viele neue Standards, und die Industrie kreiert immer wieder neue Modewörter. Jedoch hat auch die Open Source Community viel in diesen Bereichen zu bieten. In der Tat basieren viele kommerzielle Produkte auf Open Source Projekten.

Das Ziel dieses Artikels ist es, zu zeigen, dass Open Source auch im Bereich Web Services viel zu bieten hat. Dieser Artikel ist in zwei Kapitel geteilt. Das erste Kapitel zeigt Web Services und Open Source im allgemeinen. Das zweite Kapitel zeigt spezifische Open Source Werkzeuge.

) Artikel)

Orientation in Objects GmbH

Weinheimer Str. 68
D-68309 Mannheim

Tel. +49 (0) 6 21 - 7 18 39 - 0
Fax +49 (0) 6 21 - 7 18 39 - 50

www.oio.de info@oio.de

Java, XML, UML, XSLT, Open Source, JBoss, SOAP, CVS, Spring, JSF, Eclipse

KAPITEL I

EINE OPEN SOURCE PRODUKT PALETTE

Eine Produkt Palette für eine Web Service-Anwendung besteht aus einem Betriebssystem, einer Datenbank, einem Anwendungsserver, einem Webserver, einer Servlet-Engine und einer SOAP-Engine. Kommerzielle Web Service-Produkte beinhalten häufig mehrere oder alle dieser Komponenten. Von IBM zum Beispiel können Sie alles beginnend von einem Betriebssystem bis hin zu einer SOAP-Engine bekommen. Produkte wie der WebLogic Server von BEA beinhalten einen Anwendungsserver, eine SOAP-Engine und eine Business Orchestration-Engine. Open Source Gegner kritisieren, dass mehrere verschiedene Open Source Komponenten integriert werden müssen. Für jede Komponente der Palette muss ein bestimmtes Produkt aus einer Vielzahl von Varianten ausgewählt werden. Zum Beispiel, können Jakarta Tomcat oder Jetty als Webcontainer Verwendung finden. Nehmen wir an, Sie haben sich für Jakarta Tomcat entschieden, dann müssen Sie sich für eine der 47 (bezogen auf Juni 2004) verfügbaren Versionen auf der Jakarta Webseite entscheiden. So könnte Ihre Palette z.B. wie folgt aussehen:

- Redhat Linux 8.0
- MySQL 4.0
- JBoss 3.2.4
- Apache 2.0.49
- Tomcat 5.0.25
- Axis 1.1

Es ist unwahrscheinlich, dass jemand anderer eine Palette mit genau denselben Programmen und Versionszahlen benutzt. Infolgedessen kann es zu einigen Kompatibilitätsproblemen kommen und Sie könnten der Einzige sein, welcher diese besondere Zusammenstellung von Komponenten benutzt. Aber die Situation ist nicht so schlecht, wie einige Open Source Gegner es uns glauben lassen möchten. Da einige Softwareprodukte und Versionen populärer sind als andere, reduziert sich die Auswahl an Produkten und Versionen erheblich. Bei der Produktion eines Produkts sollten Sie auf stabile und bekannte Versionen setzen. Open Source Software Pakete, wie zum Beispiel das JBoss Paket, welches bereits einen Anwendungsserver und einen Webcontainer enthält, machen es leichter, sich für eine Kombination zu entscheiden. JBoss bietet auch ein komplettes Paket aus Datenbank, einem Anwendungsserver, einem Webcontainer und einer SOAP-Engine an.

Bei der Open Source Entwicklung spielen offene Standards eine wichtige Rolle und Interoperabilität ist ein wichtiges Ziel. Viele Entwickler haben Open Source Komponenten im Einsatz und berichten ausführlich über die von Ihnen gefundenen Programmfehler. Somit werden Probleme früh erkannt und behoben. Im Fall einer Unklarheit bietet die riesige Entwickler-Community ihre Hilfe an.

DIE KOMBINATION AUS OPEN SOURCE UND KOMMERZIELLEN PRODUKTEN

Dank den Offenen Standards können Open Source und kommerzielle Produkte zu einer Palette integriert werden. Eine Palette könnte auch wie folgt aussehen:

- Windows 2000 Server
- Oracle Datenbank
- JBoss Anwendungsserver
- Tomcat Webcontainer

- Axis SOAP-Engine

Der Benutzer hat immer die Wahl, die besten Produkte zu verwenden - ob es nun Open Source oder kommerzielle Produkte sind. Die Produkte können sogar später ersetzt werden, wenn sie versagen, oder wenn ein besseres Produkt verfügbar wird.

INTEROPERABILITÄT

Interoperabilität ist ein wesentlicher Bestandteil von Web Services. In der Theorie sollten alle Produkte im Stande sein, miteinander zu kommunizieren. Jedoch verschlechtert sich die Situation durch die Vielfalt von Produkten und Standards sowie die Abweichungen einiger Produkte zu den Standards. Es ist unmöglich zu bestimmen, ob Open Source oder kommerzielle Software interoperabler ist. Open Source Werkzeuge werden häufig gepatcht. Diese Patches werden jedoch gewöhnlich erstellt, nachdem eine Inkompatibilität entdeckt worden ist. Vor einer Veröffentlichung mögen kommerzielle Produkte besser gegen Standards getestet sein, um wie bspw. im Falle von WS-I zertifiziert zu werden.

BENUTZERFREUNDLICHKEIT

Open Source Projekte erinnern an text-basierte Abenteuerspiele - die Dokumentation ist häufig über mehrere Orte verstreut. Die Projektdokumentation ist häufig zu kurz und mangelhaft. Wertvolle Informationen sind im Quellcode begraben und sind nur für erfahrene Programmierer zugänglich. Jedoch bietet die Community um das Open Source Projekt eine reichhaltige Quelle an Wissen. Die Archive der Foren und Mailing Listen sind mit Information vollgestopft. Auf neue Fragen wird gewöhnlich sofort geantwortet.

Kommerzielle Software beinhaltet häufig eine ausgezeichnete Dokumentation und schrittweise geführte Tutorials. Im Gegensatz zu Open Source Produkten haben fast alle kommerziellen Web Service-Produkte benutzerfreundliche Graphikschnittstellen und Wizards. Die Lernkurve ist nicht so steil. Somit kann ein integriertes Produkt innerhalb weniger Tagen erstellt werden.

Obwohl Dokumentation und Unterstützung für die Open Source und kommerziellen Produkte im Allgemeinen verschieden sind, ist es nicht möglich zu sagen, dass ein Produkt besser ist als das Andere. Bevor Sie sich für ein bestimmtes Open Source Produkt entscheiden, sollten sie zuvor Dokumentationen, Bücher, Newsgroup und Community Beiträge, die zu diesem Produkt existieren, untersuchen.

KAPITEL II: OPEN SOURCE PROJEKTE

Dieses Kapitel beschreibt Web Service-Werkzeuge im Open Source Bereich. Die im Folgenden aufgeführten Werkzeuge zeigen einige typische Verdächtige ohne Anspruch auf Vollständigkeit

SOAP-ENGINES

Eine SOAP-Engine ist ein Server oder eine Programmbibliothek, welche die Kommunikation über das SOAP-Protokoll ermöglicht. Es gibt viele Open Source SOAP-Engines für alle möglichen Programmiersprachen und Plattformen. Für einige Sprachen sind sogar nur Open Source Engines verfügbar.

APACHE AXIS

Apache Axis ist eine SOAP-Engine, welche der liberalen Lizenz der Apache Software Foundation unterliegt. Axis kann für die Entwicklung von Clients und Servern und auch für die Entwicklung anderer SOAP-basierter Werkzeuge wie Web Services Gateways und Firewalls Verwendung finden. Axis unterstützt die Java Standards JAX-RPC und SAAJ. Enterprise JavaBeans können als Web Services eingebunden werden. Mehrere kommerzielle und nicht-kommerzielle Produkte basieren oder verwenden Axis um Web Services zu unterstützen. Als Beispiele gelten Apple's WebObjects, Borland's JBuilder, Macromedia's JRun und Coldfusion.

SPHEON JSOAP

JSOAP ist eine kleine und schnelle SOAP-Engine für Java von Florian Müller. Das Werkzeug bietet anpassbare Serializer, begrenzte WSDL Unterstützung und einen graphischen WSDL Wizard.

GSOAP

gSOAP ist ein Toolkit für den Gebrauch und die Entwicklung von Web Services in C und C++. Damit kann Funktionalität aus nativen Bibliotheken und Legacy-Anwendungen als Web Service veröffentlicht werden. gSOAP ist kompatibel mit dem WS-I Basic Profil 1.0a und informiert über potentielle Interoperabilitätsprobleme während der Compilierungszeit. Das Toolkit benutzt spezielle XML parsing Techniken, und ist somit sehr schnell. Sogar auf langsamer Hardware ergaben sich Roundtripzeiten von unter drei Millisekunden - in diesem Bereich gibt es fast keinen Unterschied zu CORBA. Der Speicherbedarf von Web Service-Clients und Servern, die gSOAP verwenden, kann ziemlich klein sein (weniger als 150 KBytes). gSOAP kann in die Webserver Apache HTTPD und Microsoft IIS integriert werden. Alternativ kann der enthaltene Web Server verwendet werden. Damit es zu einer sicheren Verbindung kommt, kann gSOAP über HTTPS kommunizieren. An einer Unterstützung für WS-Security wird zur Zeit gearbeitet.

SICHERHEITSLÖSUNGEN

In der Zeit vor SOAP wurden Nachrichten noch mit Hilfe komplizierter Protokolle und unverständlichen Inhalten ausgetauscht. Verschlüsselung und Unterschriften wurden nicht immer benötigt, da Nachrichten nur schwer zu dechiffrieren oder zu verändern waren. SOAP-Nachrichten sind XML Dokumente, die in einem gewöhnlichen Texteditor geöffnet werden können. Verglichen mit CORBA Nachrichten, die das IIOP-Protokoll verwenden sind SOAP-Nachrichten für Menschen viel leichter zu erfassen. Zusätzlich können SOAP-Nachrichten mit Hilfe einfachster Werkzeuge modifiziert werden. SOAP-Nachrichten sind eine folglich eine Einladung für Hacker - vor allem wenn sie ungeschützt durch das Internet verschickt werden. Folglich benötigt eine SOAP-Kommunikation mehr Sicherheitsmassnahmen als frühere Protokolle.

Auf den ersten Blick mag es sonderbar erscheinen, eine Quell-Offenen Sicherheitslösung in Erwägung zu ziehen. Jedoch werden auch hier Sicherheitslücken schneller entdeckt und repariert, wenn der Quellcode der Komponente frei verfügbar ist.

WSS4J

Mit WSS4J entwickelt man bei Apache eine OASIS Web Services Security (WS-Security) Implementierung als Subprojekt einer "Web Services Functionality Extension" (WSFX). WSS4J benutzt hierzu Axis und XML-Security.

OPENSAML

SAML ist ein OASIS-Standard, der Single Sign On für Web Services ermöglicht. SAML definiert das Format von XML Nachrichten mit Authentifizierungs- und Autorisierungsdaten. openSAML enthält mehrere Java und C++ Bibliotheken, welche das Parsen, das Erzeugen und den Transport von SAML Nachrichten ermöglichen. Internet2 Mitglieder haben openSAML als ein Teil ihrer Arbeit an der "Shibboleth federation solution" benutzt.

SOURCEID

Die Ping Identity Corporation bietet seine Federation Gateway Software unter einer beschränkten Open Source-Lizenz an. SourceID ermöglicht Identity Federation, Single Sign On und Cross Boundary Security. Version 1.0 unterstützt die Standards von SAML 1.0 und Liberty 1.1. Die nächste Version 2.0 wird SAML 1.1, Liberty 2.0, OASIS Web Services Security, WS-Trust und WS-Federation unterstützen. Implementierungen sind für Java und .NET verfügbar. SourceID kann auf Apache Tomcat, BEA WebLogic, IBM WebSphere und JBoss betrieben werden.

APACHE XML SECURITY

Das Apache-Projekt XML Security stellt eine Implementierung für die W3C Standards XML Encryption und XML Signature zur Verfügung. Eine XML Security-Bibliothek ist bereits verfügbar und XML Encryption ist in der Beta-Phase. Eine Unterstützung für XML Key Management wird geplant.

WERKZEUGE

Web Services beruhen auf XML, somit muss eine Web Service-Software auch XML bearbeiten können. XML-Nachrichten müssen geparkt werden, und Dokumente transformiert werden. Die grundlegenden Werkzeuge zum Parsen und Transformieren sind XML Parser wie Xerces und XSLT-Prozessoren wie Xalan. Viel von dieser XML Infrastruktur besteht aus Open Source Projekten, und viele kommerzielle Verkäufer verwenden diese Infrastruktur, um ihre Software zu bauen.

JUDDI

jUDDI ist die Implementierung von Apache der Universal Description, Discovery and Integration (UDDI) Spezifikation. Es benötigt eine relationale Datenbank mit einer JDBC Unterstützung und einem Servlet 2.3 konformen Webcontainer. jUDDI kann leicht in vorhandene Authentifizierungssysteme integriert werden.

CLADONIA EXCHANGER XML BROWSER

Der Exchanger XML Browser ist ein Programm zur Visualisierung von XML Dokumenten. XML-Elemente können mit einem Service verbunden werden, der ein XML-Element in einer interaktiven Graphik darstellen kann. Ein SOAP Envelope kann zu einem Envelope Image gerendert werden. Ein Klick auf das Image öffnet den entsprechenden Web Service.

APACHE WEB SERVICES FX

Die Web Services Functionality Extensions bestehen aus mehreren Subprojekten, welche WS-* Spezifizierungen implementieren. Das Addressing-Subprojekt befasst sich mit WS-Addressing; Sandesha benutzt WS-ReliableMessaging und das oben erwähnte WSS4J ist eine Erfüllung der Web Services Security auch bekannt als WS-Security.

ZUSAMMENFASSUNG

Der Softwareentwickler profitiert von der Vielzahl an kommerziellen und Open Source Produkten. Open Source Projekte für Web Services bieten ähnliche Vorteile und Nachteile wie andere Open Source Projekte. In der Vergangenheit haben Open Source Projekte für Web Services weniger Anerkennung erhalten, als sie verdienen. Aber vielversprechende Projekte für die Web Service-Sicherheit könnten das in Zukunft ändern.

REFERENZEN

- Leveraging Open Source for Web Services Development
Peltz, Chris , Hewlett Packard Company; Rogers, Claire , Hewlett Packard Company
2003-05-01
[TheServerSide.COM \(http://www.theserverside.com/tss\)](http://www.theserverside.com/tss)
- Web Services and Open Source
Manes, Anne Thomas
<http://www.itconversations.com/shows/detail2.html>
- Open Source Option for Developing Web Services
Krill, Paul , InfoWorld
<http://www.computerworld.com/printthis/2002/0,4814,76674,00.html>
- Open-Source Firm Dips Into Services SOUP
<http://news.com.com/2100-1001-253885.html?legacy=cnet>
- Java Web Services mit Apache Axis
http://www.amazon.de/exec/obidos/ASIN/3935042574/qid=1088025819/sr=1-1/ref=sr_1_8_1/028-2539436-1936511
- OpenSAML 1.0 - An Open Source Security Assertion Markup Language Implementation
<http://www.opensaml.org/>
- Web Services Project @ Apache
<http://ws.apache.org/>
- Apache XML Security
<http://xml.apache.org/security/index.html>
- Spheon JSOAP
<http://soap.fmui.de/index.php>
- gSOAP: C/C++ Web Services and Clients
<http://gsoap2.sourceforge.net/>
- SourceID Open Source Federated Identity Management
<http://www.sourceid.org/>
- Exchanger XML Browser
<http://sourceforge.net/projects/xngr/>