



webMethods 9: Umfangreiche Modellpflege

Dienstleistungsgesellschaft

Bernhard Steppan

Die Software AG schickt die neunte webMethods-Ausgabe mit Neuerungen zu den Hype-Themen der Zeit ins Rennen. Nun soll die Integrationsplattform mit Cloud Computing, Big Data und mobilen Anwendungen klarkommen.

Befragt man IT-Architekten aus verschiedenen Firmen nach der richtigen Integrationsstrategie, löst diese Frage in der Regel kontroverse Diskussionen aus. Das Spektrum der technischen Optionen reicht von selbst entwickelten SOA-Anwendungen, diversen Frameworks und quelloffenen Enterprise Services Buses (ESB) bis zu umfangreichen kommerziellen Produkten. Zu letzterer Spezies gehört die Plattform webMethods, die nun in der Version 9 vorliegt.

Seine Ursprünge hat das Produkt in der Webentwicklung. Namensgeber ist die US-Firma webMethods, 1996 gegründet und 2007 von der Software AG für 546 Millionen US-Dollar übernommen. Die Suite stellt eine Fülle von Werkzeugen, Adaptern und Servern zur Verfügung. Dazu gehören ein Integration Server, ein Broker, das Entwicklungswerkzeug Designer, Software zur Großrechnereinbindung (EntireX und ApplinX) sowie Konnektoren zu Standardanwen-

dungen, ein zentrales Administrations-tool und die SOA-Registry CentraSite – um nur die wichtigsten Bestandteile zu nennen (Abbildung 1). Weiterhin gibt es eine Geschäftsprozessmanagement-Komponente, die hier nicht näher betrachtet wird.

Es gibt mehrere Ansätze zur Systemintegration (siehe Kasten „Komplizierte Beziehungen“). Früher galt webMethods als ein typischer Vertreter des Hub-and-Spoke-Verfahrens. Im Laufe der Jahre kamen weitere hinzu. Heute bilden webMethods Integration Server und der Broker Server den Kern des ESB, der weitaus komplexere als die klassischen Hub-and-Spoke-Installationen ermöglicht.

Der Integration Server ist in Java geschrieben und enthält einen eingebetteten Tomcat/JBoss-Webserver, der sich zur Lastverteilung mehrfach starten lässt. Er beherrscht die gängigen Kommunikationsstandards: die Basisprotokolle HTTP, HTTPS, FTP, SMTP, die Netzwerkprotokolle SOAP, XML RPC sowie das Nachrichtenformat MIME beziehungsweise S/MIME. Als Datenformat kommen XML oder ein selbstdefiniertes Infrage, etwa durch Strichpunkte separierte Datensätze.

Viele Verbindungen zur Außenwelt

Es existieren Konnektoren für PeopleSoft, Remedy, Salesforce, SAP, Siebel sowie zu Midrange- (IBM i) und Großrechnersystemen (z/OS). Mit den aufgeführten Protokollen, den Adaptern und Zusatzkomponenten deckt der Integration Server das Gros der Möglichkeiten über synchrone Kommunikation ab.

Den Broker verwendet man hingegen im Regelfall, um Anwendungen asynchron anzubinden. Mit seiner Hilfe lassen sich die gängigen JMS-Konzepte Point-to-Point und Publish-Subscribe umsetzen. Er kann mit mehreren Integration Servern zum Aufbau komplexer Architekturen zusammengeschaltet werden, um die Performance und die Zuverlässigkeit bei hoher Gesamtlast deutlich zu verbessern. Statt des proprietären Brokers darf der Entwickler auch einen standardkonformen JMS-Broker wie Apaches ActiveMQ einsetzen.

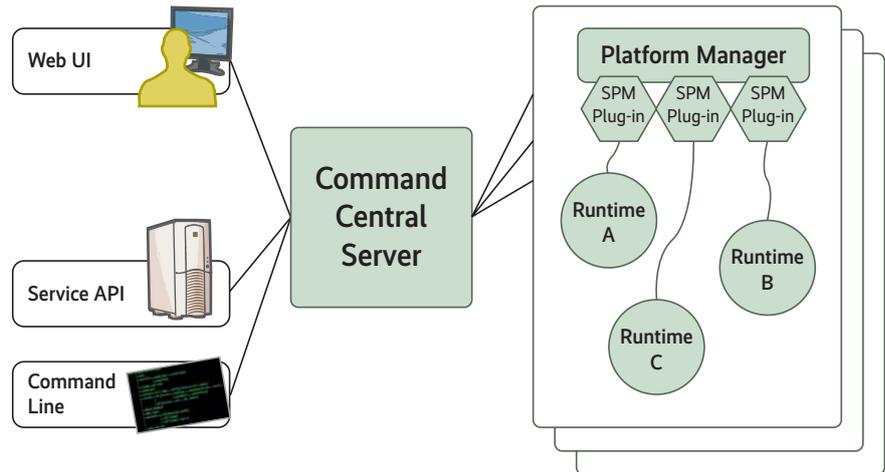
Zum Programmieren bietet webMethods eine Java-IDE auf Eclipse-Basis an. Der Designer verfügt über zahlreiche Views sowie diverse Perspektiven auf die verschiedenen Bestandteile der Suite. Mit der IDE kann der Benutzer Webservices

auf Basis des Integration Server mit unterschiedlichen Sprachen implementieren (etwa Java, C oder Flow). Das Installieren und Testen der Services im Integration Server kostet nur wenig Mühe. Einziges Manko: Die Software AG hat es immer noch nicht geschafft, ihre IDE auf die aktuelle Eclipse-Version 4.3 zu heben. Der Designer basiert weiterhin auf dem veralteten Eclipse 3.6 und ist – im Gegensatz zur originalen IDE – nicht für Mac OS X verfügbar.

Designer: Viele kleine Helferlein

Ansonsten gibt es vorwiegend Erfreuliches vom Designer zu berichten. Eine Fülle von Assistenten hilft beim Schreiben eines Service. Will der Entwickler den Dienst in Java umsetzen, legt er ihn innerhalb der IDE auf dem Integration Server ohne Verteilmechanismus an. Klickt er danach in der Integration-Server-View auf das erzeugte Template, öffnet die IDE das Codefragment in einem Editor, der die vom Assistenten angelegten Codeblöcke vor dem Überschreiben schützt. Wenngleich einige Dinge anfangs etwas ungewohnt erscheinen, profitiert der Entwickler von der vergleichsweise einfachen Handhabung. Er kann den Service nicht nur sofort ausführen und testen, sondern auch, wie bei einem Standard-Java-Dienst üblich, beliebig mit Java-Bibliotheken ergänzen.

Anders als Java-Services werden Flow-Dienste ausschließlich grafisch erstellt. webMethods Flow ist eine proprietäre

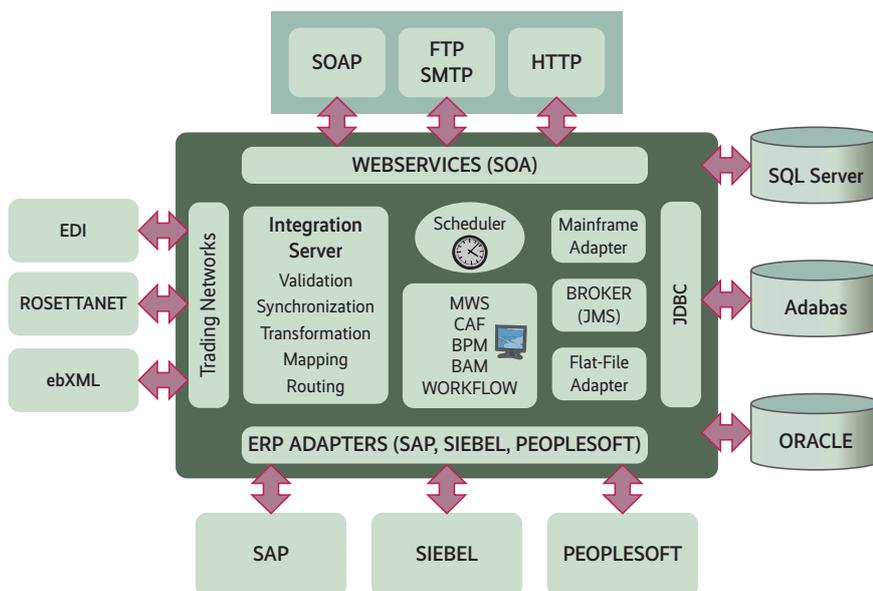


Command Central führt die in den Vorgängerversionen verstreuten Verwaltungsinstanzen unter einer einheitlichen Oberfläche zusammen (Abb. 2).

grafische Sprache, vergleichbar mit der Prozessbeschreibungssprache BPEL. Flow eignet sich für das modellgetriebene Zusammenklicken von Services ohne Codierung. Dafür existiert ebenfalls eine spezielle Perspektive mit einem Editorfenster. Es besteht aus mehreren Registerkarten mit verschiedenen Ansichten auf den Flow-Dienst. Es gibt eine Baum- sowie eine Layoutseite, eine Seite für Input-/Output-Ströme sowie eine für das Anlegen von Feldern. Die Baumansicht verschafft einen Überblick über die Prozessschritte, die der Entwickler in der Layoutansicht hinzufügt und miteinander verschaltet. Der Editor für Input-/Output-Ströme dient dazu, die Endpunkte des Service mit Ein- und Ausgabeparametern zu definieren. Wie beim Java-Pendant sind die Flow-Dienste ohne Verteilung

direkt auf dem Integration Server ausführbar.

Um einen Client für einen Service auf Basis des Integration Server zu erzeugen, reicht es aus, sich vom Designer einen Rahmen anlegen zu lassen und ihn den eigenen Wünschen anzupassen. Leider „vergisst“ der Designer immer noch, die Laufzeitkonfiguration für den Client mit allen notwendigen Programmierbibliotheken zu erzeugen. Folge: Der Start des Clients endet mit einem Laufzeitfehler. Man muss die notwendigen Bibliotheken selbst zusammensuchen und die Konfiguration anpassen. Da der Designer eine vollwertige Eclipse-IDE ist, kann man mit ihm grafische Oberflächen für Rich Clients mit RCP oder Swing wie gewohnt erstellen. Will man Webclients entwickeln, die der Integration Server ohne Umwege ausführen kann, gilt die Einschränkung, dass webMethods intern nur JavaServer Faces 1.2 unterstützt. Das bedeutet, dass ein auf JSF 2.x basierender JSF-Client separat von der webMethods-Infrastruktur entwickelt und betrieben werden sollte.



webMethods stellt zahlreiche Integrationstechniken, -werkzeuge und Adapter zum Einbinden von externen Rechnern und Standardsoftwareprodukten zur Verfügung (Abb. 1).

Mainframes mit Beharrungsvermögen

Großunternehmen arbeiten häufig noch mit altgedienten Legacy-Programmen, die zwar nicht mehr dem Stand der Technik entsprechen (etwa auf Basis von Adabas-Natural oder COBOL/DB2), sich jedoch nur schwer ablösen lassen. Beim Einbinden von Mainframe-Anwendungen in eine serviceorientierte Systemlandschaft hilft das Integration Server Add-on EntireX. Es beherrscht z/OS 1.1, 1.2, 1.13, z/VSE 4.2, 4.3, IBM i sowie die Unix-Systeme AIX 6.1, 7.1, SUSE und Red Hat

Komplizierte Beziehungen

Innerhalb einer Point-to-Point-Architektur kann jeder mit jedem reden. Vorteil: schnelle Kommunikation und kaum Overhead. Wenn kein Message Broker im Spiel ist, entstehen jedoch starke Bindungen zwischen den teilnehmenden Systemen. Selbst gestrickte Webservices auf Point-to-Point-Basis führen daher ohne gute Dokumentation der Servicekontrakte schnell ins Chaos.

Bei der Hub-and-Spoke-Architektur steht ein Server (Hub) im Zentrum, zu dem alle Anwendungen, die zusammenarbeiten sollen, Beziehungen (Speichen) haben. Ein zentrales

System lässt sich leicht administrieren, neu angeschlossene Anwendungen sind sofort mit dem Hub verbunden. Nachteilig wirken sich die oftmals auftretenden Performanceeinbrüche aus.

In der Bus-Architektur sind alle Anwendungen lose über einen Bus verbunden. Diese Konstruktion skaliert im Allgemeinen besser als die beiden anderen, ist jedoch komplexer und schwieriger zu administrieren. Beim Publish/Subscribe Pattern, einer Ausprägung des Bus-Konzepts, abonnieren mehrere Services eine Nachricht.

Linux für IBMs System z/OS. Mit EntireX kapselt der Entwickler beispielsweise im IMS und CICS laufende COBOL- und C-Programme. Diese Aufgabe erledigt er ebenfalls mit dem Designer, der ihm dazu eine EntireX-Perspektive anbietet.

Großrechneranwendungen schöner machen

Mit ApplinX ist es beispielsweise möglich, 3270-Masken in Java-Weboberflächen umzuwandeln. Die zeitaufwendige manuelle Arbeit, die normalerweise hinter solchen Aktionen steckt, reduziert sich durch das Einlesen von Screen-Definitionen wie IBMs CICS Basic Message Service oder Screen Definition Facility. Der Code der Legacy-Anwendung bleibt dabei unberührt. Laut Hersteller stehen über 40 Assistenten für ApplinX im Designer zur Verfügung.

Die Übernahme von Terracotta in die webMethods-Suite ist sicher die spektakulärste Neuerung. Im Mai 2011 hatte die Software AG das gleichnamige Unternehmen gekauft. Es wurde mit dem Caching-Produkt Ehcache bekannt, das seinen Dienst in O/R-Mapping-Frameworks wie Hibernate verrichtet. Terracotta verschafft einer webMethods-Anwendung mehr Speicher, als es die Standard-Java-Methoden vermögen. Das Werkzeug erweitert den limitierten Java-Heap um einen au-

ßerhalb liegenden Cache. Abhängig von der verwendeten Plattform schaufelt dieser technische Kunstgriff bis zu einem Terabyte an zusätzlichem Cache frei. Angeblich führt das bei Anwendungen, die mit großen Datenmengen operieren, beispielsweise zu einer über hundertmal schnelleren Reaktionszeit eines Warenkorbs.

Command Central ist eine weitere neue Komponente. Wen immer schon genervt hat, dass man die Bestandteile der Suite an unterschiedlichen Stellen konfigurieren und überwachen musste, bekommt nun ein übergreifendes Verwaltungswerkzeug. Es besteht aus einer Administrationsoberfläche, einem zentralen Server und einem oder mehreren mit den verschiedenen Laufzeitumgebungen verbundenen Plattformmanagern. Über die Admin-Konsole lassen sich die Kommandozeilenparameter für alle vorhandenen Module festlegen. Der Plattformmanager reicht sie an die verschiedenen Laufzeitumgebungen weiter (Abbildung 2). Das Ganze sieht komplex aus, besonders da es sowohl für den Server als auch für die Plattformmanager eigene Remote-Java- und REST-APIs gibt. Es drängt sich die Frage auf, warum man sich nicht entschieden hat, die Suite mittels des bewährten Industriestandards JMX (Java Management Extensions) zu verwalten.

Viele Adapter serienmäßig

SaaS-Angebote mit den internen IT-Systemen auf Basis von webMethods zu integrieren, ist aufgrund des Austauschs streng vertraulicher Daten nicht einfach. In der Vergangenheit waren solche Projekte zeitaufwendig und teuer. Daher hat die Software AG Konnektoren für externe Systeme wie Salesforce in die Suite eingebaut. Sie firmieren unter dem Namen CloudStreams und sollen die Benutzer in

die Lage versetzen, Integrationsvorhaben, die früher Monate in Anspruch nahmen, in wenigen Tagen abzuwickeln.

Beim Aufbau einer serviceorientierten Anwendungslandschaft ist es wichtig, zu dokumentieren, welche Services in welchen Versionen und Ausprägungen überhaupt existieren. CentraSite ist ein SOA-Registry und -Repository, mit dem sich Eckdaten der Services speichern und per UDDI (Universal Description Discovery and Integration) auslesen lassen. Die Software gibt es als kostenfreie Community- sowie als kommerzielle Variante (siehe „Alle Links“). Letztere verfügt über zusätzliche Features wie Logging und Lifecycle Management. Beide Ausführungen sind via Designer in die Suite eingebunden. Wer einen Service entwickelt und Zusatzfunktionen benötigt, wechselt in die CentraSite-Perspektive und sucht dort nach vorhandenen Angeboten, die er mit dem eigenen Service verknüpfen kann.

Bislang konnte webMethods in Sachen mobile Anwendungen kaum punkten. Mit der Mobile Suite versucht Version 9 diese Lücke zu schließen. Mit dem schon aus der Vorgängerversion bekannten Mobile Designer lassen sich neben Android-, BlackBerry- und iOS- auch HTML-5-Clients entwickeln. Hervorzuheben ist der Cross-Development-Ansatz: Der Entwickler schreibt seine mobilen Anwendungen in Java und lässt sie danach vom Cross Compiler des Mobile Designers in nativen Code übersetzen. Es soll zudem möglich sein, sie so sicher mit bestehenden IT-Systemen zu verbinden, dass nur autorisierte Apps Zugriff auf interne Systemteile bekommen.

Fazit

webMethods besteht aus einer Mischung offener und proprietärer Konzepte, die auf einer Standard-Java-Infrastruktur aufsetzen. Die Suite bietet die Möglichkeit, mit einfachen Architekturen zu starten und sie bei Bedarf zu erweitern. Release 9 markiert keine Zäsur, sondern schließt Lücken in Bezug auf Cloud Computing, Big Data und mobile Clients – alles in allem eine gelungene Modellpflege. (jd)

Bernhard Steppan

arbeitet bei der SYRACOM Consulting AG in Wiesbaden als Leading Consultant.

Alle Links: www.ix.de/ix1308080

-Wertung

- ⊕ Integrationsmöglichkeiten
- ⊕ Serviceentwicklung
- ⊕ Skalierungsoptionen
- ⊖ Designer auf Basis von Eclipse 3.6