

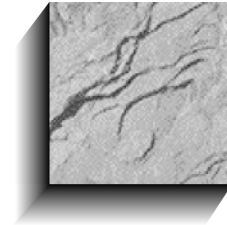
CARL HANSER VERLAG

Paul Dorsey, Peter Koletzke

Oracle JDeveloper 3 Handbuch
Der Einstieg in die Java-basierte Webentwicklung
JDeveloper Version 3.2

3-446-21818-1

www.hanser.de



Vorwort

There is something very powerful here which I think many of us are missing completely...

—kirk, JDeveloper Forum, 2001

Anlässlich der Oracle Open World Conference im Herbst 1999 nahm ich an einer der vielen technischen Konferenzen teil. Der Redner war eines der bekanntesten und erfahrensten Mitglieder der Oracle Tools Community: Paul Dorsey. Bekannt dafür, dass er kein Blatt vor den Mund nimmt, fasste er die drei letzten Releases meines Produkts wie folgt zusammen: „JDeveloper 1.0 – nutzlos. JDeveloper 2.0 – nutzlos. JDeveloper 3.0 – jetzt wird es interessant!“ Uff. Ich war mir nicht ganz sicher, ob ich wegen der beiden ersten Aussagen den Raum verlassen oder auf Grund der dritten Aussage meinen Kopf heben sollte.

Aus der Perspektive eines Betrachters, der die Ausgereiftheit und Produktivität von traditionellen Client-Server-Tools wie Oracle Forms, PowerBuilder und VisualBasic kennt, wusste ich, dass Pauls Aussage bezüglich der ersten beiden Releases des Produkts absolut korrekt war. Vor 3.0 konnte man JDeveloper als eine von vielen Java IDEs (Integrated Development Environment) abtun. Trotz der innovativen und leistungsfähigen Funktionalitäten für Java-Programmierer (wie dem schnellsten Debugger am Markt und die einmalige Code-Catch-Funktion, die suboptimale Java-Codes abfängt und korrigiert), trugen die ersten Versionen von JDeveloper wenig dazu bei, den Aufbau von mehrschichtigen (Multi-Tier) Datenbankapplikationen zu vereinfachen. Dieser Umstand war jemandem wie Paul sicher nicht entgangen.

Glücklicherweise entging Paul allerdings auch nicht der Umstand, dass sich JDeveloper 3.0 im Vergleich zu seinen Vorgängern fundamental verändert hat. Die Einbindung von Geschäftskomponenten für Java – das neue, komponentenbasierte Applikationsgerüst – sorgte dafür, dass das Urteil von „nutzlos“ in „interessant“ umschlug. So konnten wir unsere Vision einer *leistungsfähigen* Programmierumgebung und eines *produktiven* Applikationsgerüsts verwirklichen.

Ein Jahr später wurden Gerüst und Deployment-Umgebung von JDeveloper 3.0 noch einmal gründlich überarbeitet. Meines Erachtens bewegt sich das Produkt heute in einem Bereich, der weit mehr als „interessant“ ist.

Bevor Sie sich selbst in die Materie vertiefen, erlauben Sie mir, hier einige zentrale Punkte hervorzuheben, die JDeveloper und Geschäftskomponenten in Verbindung mit Java so unwiderstehlich machen:

- **Standardplattform** Das Gerüst basiert ausschließlich auf den standardmäßigen Internet-Protokollen, Sprachen und APIs. Java selbst ist immens populär und eine leistungsfähige, moderne, objektorientierte Sprache, die ausschließlich auf das Internet zugeschnitten ist.
- **„White Box“-Gerüst** Im Gegensatz zur Black Box der traditionellen 4GL-Engines der Client-Server-Ära bieten Komponentengerüste, die in einer standardisierten, objektorientierten Sprache wie Java implementiert sind, eine "White Box"-Umgebung. Dabei handelt es sich um eine transparente Umgebung, in der die internen Arbeitsschritte untersucht, Fehler behoben, und nötigenfalls weiter spezialisiert oder überschrieben werden können.
- **Multi-Tier, komponentenbasierte Architektur** Das Gerüst fördert eine klare logische Trennung zwischen Client-, Applikations- und Datenbankebene. Das ist insbesondere in der Internetumgebung wichtig, wo massive Durchsatzanforderungen typischerweise die unabhängige Skalierung der Applikationsschicht erfordern. Die klare Trennung ermöglicht auch die Wiederverwendung von komplexen, aufwändig zu produzierenden Geschäftskomponenten im sich ständig erweiternden Array von Internet-Clients – vom Desktop-Browser bis zum Handheld-Gerät.
- **Flexibler Einsatz** Die Ermittlung der optimalen Einsatzkonfiguration für eine mehrschichtige Internet-Applikation kann immens komplex sein. Brauche ich tatsächlich einen separaten, verteilten Objektserver? Und wenn, soll es EJB oder CORBA sein? Oder soll ich meine Anwendungslogik logisch in der gleichen Java Virtual Machine einsetzen wie meine Servlet-Engine oder Java-UI? Das Gerüst vereinfacht diesbezügliche Entscheidungen, indem es das logische Design der Anwendung von den Betrachtungen für den Einsatz trennt. Dieselben Geschäftskomponenten lassen sich in zahlreichen Konfigurationen einsetzen. Dadurch können Sie mit der einfachsten oder kostengünstigsten Konfiguration beginnen, und später problemlos auf andere Varianten umsteigen.

Dieses Buch konzentriert sich auf die zentralen Elemente von JDeveloper und viele andere Dinge. Es bietet einen erstklassigen Überblick zu den Themen, die Sie als erfolgreicher JDeveloper-Entwickler benötigen. Es zeigt die darunter liegenden Internet-Technologien, das Applikationsgerüst und die IDE, die alle Elemente aufnimmt. Und es

wurde aus der Sicht von zwei erfahrenen Experten geschrieben, die eigene Erfahrungen in der schönen neuen komponentenbasierten Welt des Internets gesammelt haben.

Wie das Zitat am Anfang des Vorworts zeigt, geht es hier um „Außergewöhnliches“. Verpassen Sie es nicht. Lesen Sie das Buch und entdecken Sie, wie Sie die Leistungsfähigkeit von JDeveloper und den Geschäftskomponenten für Java für Ihre eigenen Aufgaben nutzen.

Bill Dwight
Vice President, Java Tools
Oracle Corporation