



Eva/3 und XDEV

Java-Generatoren im Vergleich

Datum: 14.01.2010
Autor(en): Bernhard Steppan
URL: <http://www.computerwoche.de/1927012>

Rapid Application Development (RAD) funktioniert auch mit Java bestens. Das zeigen die Code-Generatoren "Eva/3" und "XDEV".

Professionelle Java-Entwicklungsumgebungen wie **Eclipse**¹ und **Netbeans**² sind nicht jedermanns Sache. Auf der einen Seite sehr flexibel, erschweren sie andererseits manchem Entwickler mit ihren vielen Einstellmöglichkeiten und ihrer generellen Ausrichtung auf technisch versierte Programmierer den Einstieg. Das war der Ansatzpunkt für eine Reihe von leicht zu bedienenden Java-Werkzeugen für das **Rapid Application Development**³ von Rich-Client-Datenbankanwendungen, von denen wir hier **Eva/3**⁴ und **XDEV**⁵ vorstellen wollen.

Soviel vorweg: Eva/3 und XDEV sollen dem Anwender eine Java-Umgebung für die visuelle Entwicklung von **Rich-Client-Anwendungen**⁶ für Datenbanken zur Verfügung stellen. Beide Produkte gehen dabei sehr ähnlich vor. Welchem Tool und welchem Framework man den Vorzug gibt, ist eher eine Geschmacksache. Mit beiden Umgebungen ist auf jeden Fall der Beweis erbracht, dass sich Rapid Application Development und Java bestens ergänzen.

Codegenerierung

Während man in der traditionellen Java-Entwicklung mit einer Vielzahl von Frameworks arbeitet und den Programmcode im Texteditor handoptimiert, wird mit RAD-Tools überwiegend visuell gearbeitet und ein Großteil des Programms mit einem Generator erzeugt. Die beiden hier vorgestellten Tools generieren Programme auf unterschiedliche Weise:

- Der Eva/3 Application Builder erzeugt während der Entwicklung eines Programms keinen Java-Sourcecode. Legt man zum Beispiel einen neuen Dialog an, speichert der Application Builder alle Einstellungen stattdessen in einer entsprechenden XML-Datei. Das Eva/3-Framework liest (interpretiert) diese Datei zur Laufzeit. Das unterscheidet das RAD-Werkzeug von einem Java-GUI-Builder, der in der Regel Sourcecode generiert, der kompiliert wird.
- XDEV speichert alle Projekteinstellungen in einer singulären Projektdatei ab, aus der die Entwicklungsumgebung die Anwendung kompiliert. Es entsteht also der übliche Bytecode, der zusammen mit der XDEV-Laufzeitbibliothek die Anwendung ergibt. Unter Windows und Mac OS X kann XDEV auch ein Executable samt JRE erzeugen, das per Doppelklick gestartet werden kann. Alternativ dazu lässt sich auch regulärer Java-Sourcecode generieren.

Eva/3 Application Builder

Das RAD-Tool Eva/3 Application Builder stammt vom deutschen Softwarehaus **Optadata.com**⁷. Die Firma ist kein Hersteller von Tools, sondern gehört zu der im Gesundheitswesen tätigen Optadata Gruppe. Eva/3 Application Builder verwendet das Softwarehaus auch selbst zur Entwicklung von Anwendungen für das Gesundheitswesen. Das **Java-Werkzeug**⁸ liegt mittlerweile in der Version 1.5.5 vor und ist in der Basisversion

zum moderaten Preis von 350 Euro erhältlich.

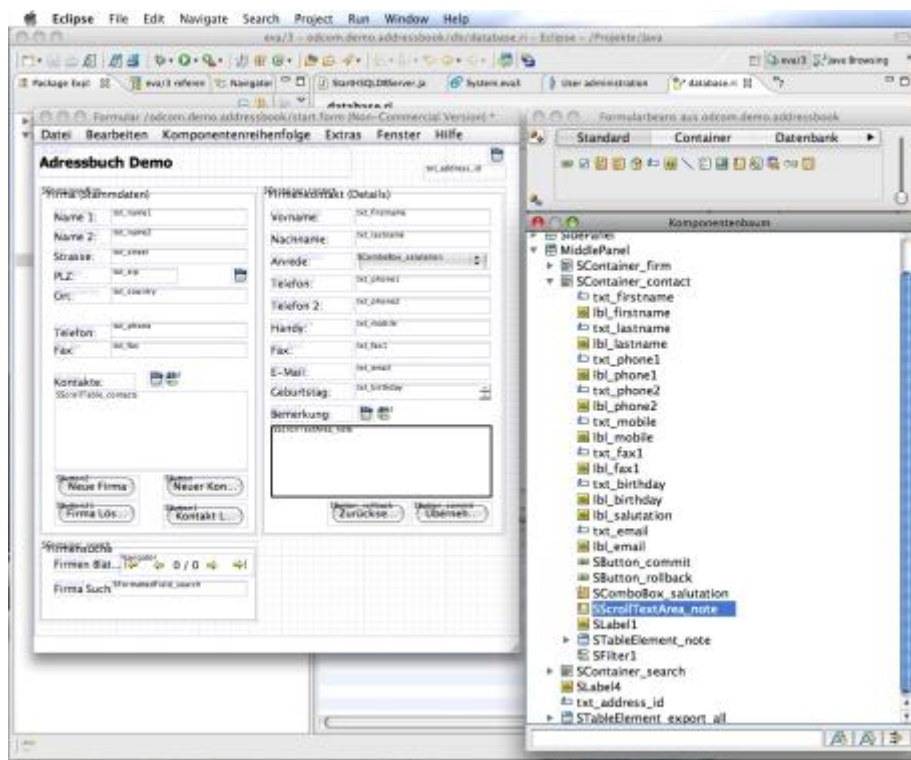
Eva/3 ist genau genommen keine vollständige Entwicklungsumgebung, sondern ein Eclipse-Plug-in. Das heißt, dass alle Werkzeuge wie Editor, **Compiler**⁹, **Debugger**¹⁰ und Projektverwaltung von der Eclipse-Entwicklungsumgebung stammen. Der Vorteil: Einem mit Eclipse vertrauten Entwickler bereitet das RAD-Tool nur wenig Einarbeitungsaufwand, da er viele gewohnte Werkzeuge weiterverwenden kann. Eva/3 Application Builder ist in Java programmiert und läuft unter **Windows**¹¹, mehreren **Linux**¹²-Betriebssystemen sowie **Mac OS X**¹³.

Eva/3-Installation

Das Plug-in kann sowohl mit einer der Standard-Eclipse-Editionen (Classic, JEE etc.) als auch mit IBMs **Websphere Studio**¹⁴ verwendet werden. Die Installation von Eva/3 verläuft wie die jedes anderen normalen Eclipse-Plug-ins. Man kopiert zum Beispiel Plug-ins und Features der Installations-CD oder der heruntergeladenen Version in die entsprechenden Eclipse-Ordner oder verwendet die Softwareaktualisierung von Eclipse. Nach der Installation und einem Neustart der Eclipse-Entwicklungsumgebung stehen die neuen Funktionen von Eva/3 Application Builder zur Verfügung.

Das RAD-Werkzeug stellt dem Java-Entwickler eine spezielle Eclipse-Perspektive, zwei Assistenten zum Anlegen eines Eva/3-Projekts, diverse neue Sichten und ein Eva/3-Framework zur Verfügung. Da der Schwerpunkt des RAD-Tools auf der Entwicklung von Datenbankanwendungen liegt, unterstützt das Werkzeug praktisch alle wichtigen relationalen Datenbanken von **Access**¹⁵ und **SQL Server**¹⁶ über quelloffene Datenbanken wie **MySQL**¹⁷ und **HSQldb**¹⁸ bis zu Highend-Systemen wie **DB2**¹⁹ und **Sybase**²⁰.

Eva/3: Top-down-Entwicklung



Der Eva/3 Application Builder verfügt über einen integrierten GUI-Builder. Mit ihm lassen sich Formulare leicht gestalten und mit Programmlogik verbinden.

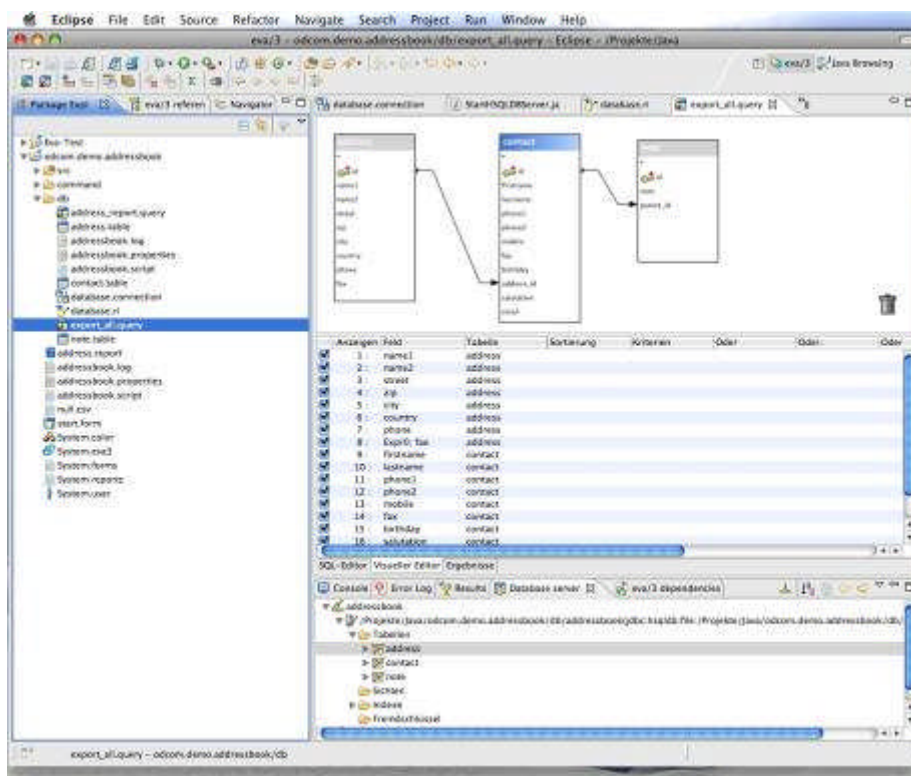
Die Entwicklung von Datenbankanwendungen verläuft bei Eva/3 top-down, also von der grafischen Oberfläche aus. Dialoge und Fenster werden bei Eva/3 generell als Formulare bezeichnet. Beim Erzeugen von Formularen zum Beispiel des Hauptfensters oder der Dialoge helfen Assistenten. Formulare lassen sich entweder manuell oder direkt aus Datenbanktabellen oder Datenbank-Views erzeugen. Wählt man das erste Verfahren aus, so arbeitet man mit einem **GUI-Builder**²¹ namens "Form Editor". Er stellt das Formular in einem eigenen Fenster dar, das von zwei Komponentenpaletten und der Darstellung des Komponentenbaums umgeben ist.

Über die Komponentenpalette zieht man sich GUI-Bausteine auf sein Formular und richtet die Bausteine danach

seinen Wünschen gemäß aus. Es gibt Standardkomponenten wie Textfelder oder Schaltflächen, Container-Komponenten wie Split Panes oder Tab Panes und Datenbankkomponenten wie Tabellen oder Filter. Damit man in verschachtelten Layouts nicht die Orientierung verliert, kann man sich im Komponentenbaum einen Überblick über die GUI-Bausteine verschaffen. Das ist notwendig, denn vom Formular selbst wird kein Java-Code erzeugt, in dem man die Komponente als Objekt entdecken könnte. Stattdessen erstellt der Form Editor eine **XML**²²-Konfiguration des Formulars, die anschließend von der Laufzeitbibliothek eingelesen wird.

Eva/3-Anwendungen

Alle Eva/3-Anwendungen basieren auf der Eva/3-Laufzeitbibliothek "eva3rt.jar", die zu jeder Anwendung hinzugefügt werden muss. Dieses Eva/3-Framework ist zwar nicht quelloffen, aber laut Herstellerangaben kann eine Sourcecode-Lizenz erworben werden. Die Laufzeitbibliothek selbst ist nicht vollständig proprietär, sondern basiert auf den **Java-Foundation-Classes**²³. Das heißt, alle Oberflächenkomponenten stammen ursprünglich von der Java-Bibliothek **Swing**²⁴. Daher lässt sich das Look and Feel der Anwendung ähnlich einer klassischen Swing-Anwendung anpassen. Hierzu setzt man in der zentralen Konfigurationsdatei den entsprechenden Wert.



Mit dem SQL-Editor des Eva/3 Application Builders kann man nicht nur SQL-Befehle absetzen. Eine eigene Ansicht stellt das Datenmodell und Beziehungen zwischen den Tabellen dar.

Auch beim Kern der Datenbankanwendung, dem Datenmodell, der Datenbankanbindung und bei der Programmierung von Datenbankabfragen kann man mit dem Application Builder sehr schnell produktiv arbeiten. Es gibt eine eigene Sicht für Datenbank-Server. Hier werden Verbindungen zu bestehenden Datenbanken hergestellt und der Tabelleninhalt angezeigt. Eva/3 unterstützt nicht nur statische, sondern auch dynamische **SQL-Abfragen**²⁵. In einem Query-Editor lassen sich SQL-Abfragen schreiben und testen. Auch bei der Arbeit mit Datenbanken wird kein Java-Code verwendet, sondern es werden Konfigurationsdateien erzeugt, die das Laufzeitsystem später einliest.

An einigen Stellen ist die Arbeit mit dem Application Builder nicht sehr intuitiv. Aber kommt man einmal mit dem Werkzeug nicht mehr weiter, hilft die ausgezeichnete Dokumentation. Insgesamt wird bei der Arbeit mit dem Application Builder und dem Studium der Handbücher schnell deutlich, dass man mit einem ausgereiften, durchdachten Tool arbeitet. Viele Tutorials scheuen nicht vor schwierigen Themen wie dem Anlegen einer Benutzerverwaltung und referentieller Integrität zurück. Selbst Einsteiger können dank dieser Hilfestellung mit dem Werkzeug schnell produktiv arbeiten.

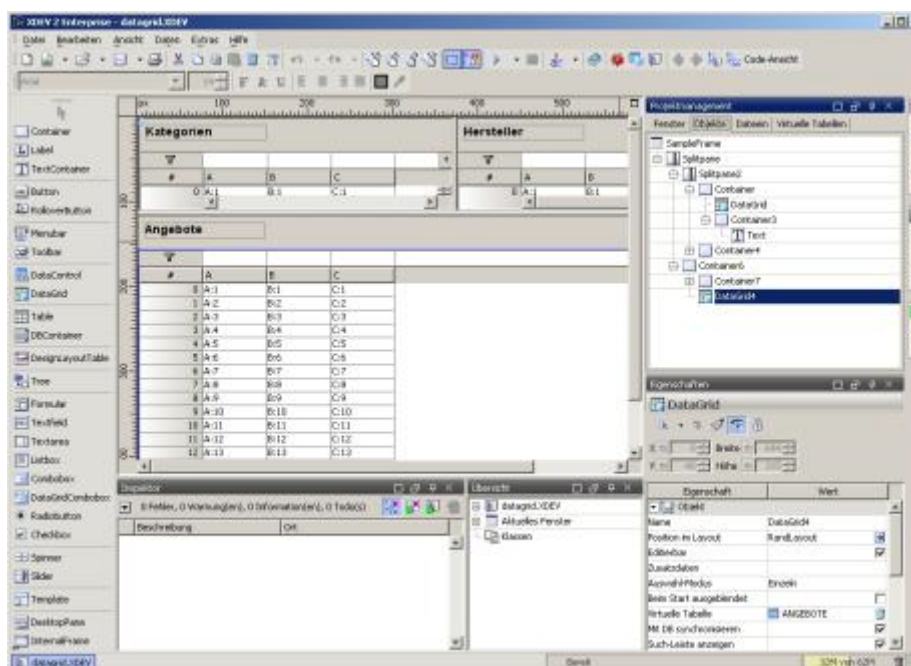
XDEV 2, Version 2.5

Im Vergleich zu Optadata folgt der amerikanische Hersteller **XDEV**²⁶ mit seinem RAD-Tool XDEV 2 etwas

anderen Grundsätzen. XDEV 2 ist eine komplett neu entwickelte Entwicklungsumgebung, bei der alle Werkzeuge wie Editor, GUI-Builder und Debugger vom Hersteller XDEV selbst stammen. Auch dieses RAD-Tool verwendet eine Laufzeitbibliothek, das XDEV Application Framework. Der Hersteller hat die Zeichen der Zeit aber erkannt und es vor einiger Zeit als Open Source freigegeben. Wer möchte, kann über eine **Community**²⁷ an der Entwicklung teilnehmen.

Das Framework nennt sich jetzt Open-XDEV und steht unter der **Common Public License**²⁸ (CPL). Das bedeutet, dass kommerzielle Anwendungen mit dem Framework entwickelt werden dürfen. Eine Lizenz der Entwicklungsumgebung ist hingegen nur für nichtkommerzielle Anwendungen kostenfrei, während die kommerzielle Version rund 1700 Euro kostet. XDEV 2 unterstützt eine breite Palette von Datenbanksystemen wie **Caché**²⁹, **DB2** und **Oracle**³⁰ und ist für die wichtigsten Betriebssysteme verfügbar.

XDEV - der Start



Mit dem GUI-Builder von XDEV lassen sich ansprechende Oberflächen in kurzer Zeit gestalten. Eine Fülle von vorgefertigten GUI-Komponenten steht hierbei zur Verfügung.

Nach dem Start der Entwicklungsumgebung befindet man sich in der GUI-Builder-Ansicht. Im oberen Bereich der IDE sind Menü und Symbolleiste untergebracht, auf der linken Seite die Komponentenpalette und in der Mitte der GUI-Builder. Auf der rechten Seite sind die Projektverwaltung und der Eigenschaften-Inspektor platziert, während sich im unteren Bereich Inspektor und ein Übersichtsfenster befinden. Wer mit der Anordnung der Fenster nicht zufrieden ist, kann alles anders anordnen, da sich alle Fenster "abreißen" und neu anordnen lassen.

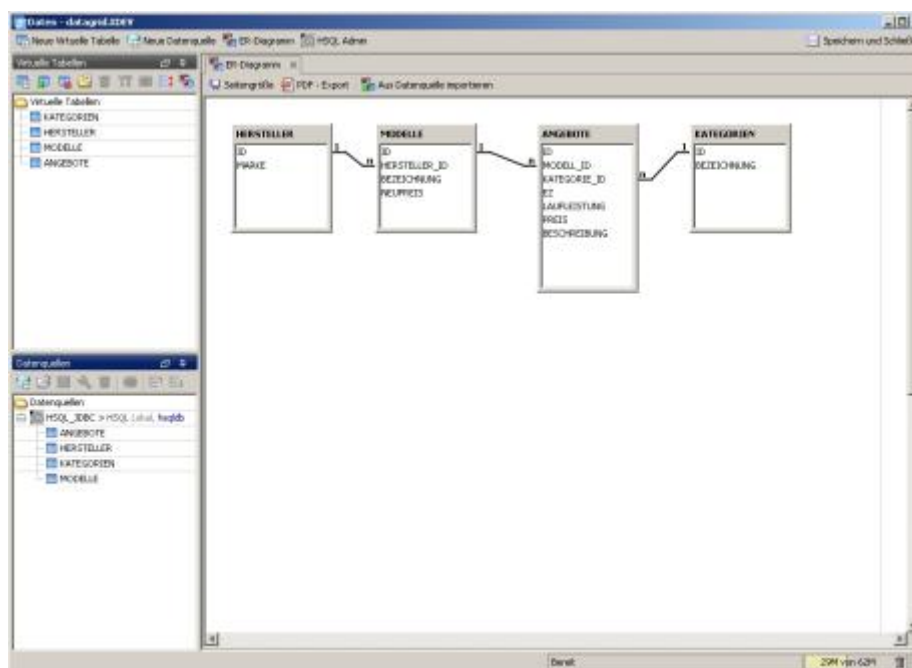
Wie bei der Konkurrenz Eva/3 erfolgt die Entwicklung von Rich-Client-Datenbankanwendungen von der Oberfläche aus. Zentraler Ausgangspunkt ist daher der GUI-Builder. Zur Entwicklung einer grafischen Oberfläche greift man auf die Komponentenpalette zurück, die viele Swing- und einige spezielle Komponenten des XDEV-Frameworks enthält. Die Komponenten können einfach per Drag and Drop von der Palette auf die Arbeitsfläche des GUI-Builder gezogen werden. Die Oberfläche einer XDEV-Anwendung basiert also auf der Java-Bibliothek Swing.

Arbeiten mit XDEV

GUI-Komponenten wie Schaltflächen oder Tabellen können auf der Arbeitsoberfläche ohne Verwendung komplizierter Java-Layout-Manager frei an einer bestimmten Position platziert werden. Diese Art der Oberflächengestaltung unterstützt der GUI-Builder mit Hilfslinien und Linealen. Will man ein typisches Java-Layout umsetzen, lassen sich aber auch die Java-Layout-Manager verwenden. Beide Arten des Oberflächendesigns werden vom GUI-Builder sehr gut unterstützt.

Für die Entwicklung von Menüs gibt es einen speziellen Menüeditor, mit dem man Menüleisten, Menüs und Menüpunkte visuell erzeugen und danach benennen kann. Die Verzahnung mit der Programmlogik erfolgt über

die Codeansicht. Wechselt man in diese Ansicht, kann man den entsprechenden Code zum Menü sofort eingeben. Um ein Menü zu testen, muss man die Anwendung nicht unbedingt kompilieren, da die XDEV-Entwicklungsumgebung über eine Vorschau verfügt. Insgesamt ist die Entwicklung einer grafischen Oberfläche mit XDEV intuitiv und ohne Studium von Handbüchern zu bewältigen.



Der Dateneditor hilft bei der Arbeit mit den XDEV-typischen virtuellen Tabellen. Eine eigene ER-Diagrammdarstellung visualisiert Entitäten und Beziehungen.

Bei der Entwicklung des Kerns einer Datenbankanwendung hilft der integrierte Dateneditor. Er erlaubt dem Anwender, ein **Entity-Relationship-Modell**³¹ (ER-Modell) zu entwerfen, ein vorhandenes zu importieren und Datenbankabfragen zu programmieren. Ein ER-Modell ähnelt hierbei stark Microsofts Access und zeigt Tabellen sowie deren Beziehungen. XDEV-Anwendungen basieren zwar auf dem im Java-Bereich üblichen **JDBC**³²-Modell. Sie verwenden aber spezielle virtuelle Tabellen als Zwischenspeicher für Abfragen. Das heißt, dass weiterhin ein JDBC-Treiber benötigt wird.

Steht das Datenmodell, muss es mit den entsprechenden GUI-Komponenten wie einer Tabellenansicht verbunden werden. Dies geschieht, indem man der GUI-Komponente im Eigenschaftenfenster die entsprechende virtuelle Tabelle zuweist. Verschiedene GUI-Komponenten des XDEV-Frameworks haben eine Reihe von Eigenschaften, die auf die Entwicklung von Datenbankanwendungen zugeschnitten sind, so zum Beispiel "Daten synchronisieren". Dadurch ist der Codierungsaufwand vergleichsweise gering.

Der XDEV-Codeeditor

Wer nicht nur einen Datenbank-Viewer schreibt, kommt aber auch bei XDEV nicht um etwas Programmierarbeit herum. Hier hilft der Codeeditor, der bei XDEV etwas aus dem Rahmen des Üblichen fällt. Das liegt daran, dass man auf die übliche Java-Syntax verzichtet hat. Der Codeeditor beherrscht stattdessen drei verschiedene Darstellungen des Sourcecodes: XDEV-Object-Script, XDEV-Basic und XDEV-NLS. Die Object-Script-Syntax ähnelt weitestgehend dem üblichen Java, während sich der Basic-Dialekt am Vorbild von Microsoft orientiert. Wählt man XDEV-NLS, so werden Schlüsselwörter in Deutsch angezeigt.

Der Codeeditor dürfte für kontroverse Diskussionen unter Java-Entwicklern sorgen. Er ist auf jeden Fall nicht mit den Vorbildern **Eclipse**³³ oder **Netbeans**³⁴ zu vergleichen und erfordert eine gewisse Umstellung. Das hängt schon allein mit der ungewohnten Syntax zusammen. Dafür hat der Hersteller auch hier wieder versucht, dem Einsteiger alles so leicht wie möglich zu machen. Für ihn ist die Hürde, mit XDEV produktiv zu arbeiten, auf jeden Fall geringer als mit den klassischen Java-Entwicklungsumgebungen. (ue)

Links im Artikel:

¹ <http://www.eclipse.org/>

² <http://netbeans-forum.de/>

- 3 http://de.wikipedia.org/wiki/Rapid_Application_Development
- 4 <http://www.optadata.com/index.php?page=2001>
- 5 <http://www.xdev-software.de/download/login.php>
- 6 http://www.google.de/search?hl=de&defl=de&q=define:Rich-Client&ei=TqZMS8yMNdKb_Ab_gtifDg&sa=X&oi=glossary_definition&ct=title&ved=0CAoQkAE
- 7 <http://www.optadata.com/>
- 8 http://de.wikipedia.org/wiki/Java_%28Programmiersprache%29
- 9 <http://de.wikipedia.org/wiki/Compiler>
- 10 <http://de.wikipedia.org/wiki/Debugger>
- 11 <http://www.microsoft.com/Germany/windows/>
- 12 <http://www.linux.de/>
- 13 <http://www.apple.com/de/macosex/>
- 14 <http://www-01.ibm.com/software/de/websphere/ws-studio.html>
- 15 http://de.wikipedia.org/wiki/Microsoft_Access
- 16 <http://technet.microsoft.com/de-de/sqlserver/default.aspx>
- 17 <http://www.mysql.de/>
- 18 <http://hsqldb.org/>
- 19 <http://www-01.ibm.com/software/de/data/db2ims/db2ud.html>
- 20 <http://www.sybase.de/>
- 21 http://wiki.computerwoche.de/doku.php/programmierung/gui-builder_fuer_eclipse
- 22 http://de.wikipedia.org/wiki/Extensible_Markup_Language
- 23 http://de.wikipedia.org/wiki/Java_Foundation_Classes
- 24 http://de.wikipedia.org/wiki/Swing_%28Java%29
- 25 <http://de.wikipedia.org/wiki/SQL>
- 26 <http://www.xdev.com/>
- 27 <http://www.openxdev.org/>
- 28 http://de.wikipedia.org/wiki/Common_Public_License
- 29 <http://www.intersystems.de/cache/index.html>
- 30 <http://www.oracle.com/lang/de/database/index.html>
- 31 <http://de.wikipedia.org/wiki/Entity-Relationship-Modell>
- 32 http://de.wikipedia.org/wiki/Java_Database_Connectivity
- 33 <http://www.eclipse.org/>
- 34 http://netbeans.org/index_de.html

IDG Business Media GmbH

Alle Rechte vorbehalten. Jegliche Vervielfältigung oder Weiterverbreitung in jedem Medium in Teilen oder als Ganzes bedarf der schriftlichen Zustimmung der IDG Business Media GmbH. DPA-Texte und Bilder sind urheberrechtlich geschützt und dürfen weder reproduziert noch wiederverwendet oder für gewerbliche Zwecke verwendet werden. Für den Fall, dass in Computerwoche unzutreffende Informationen veröffentlicht oder in Programmen oder Datenbanken Fehler enthalten sein sollten, kommt eine Haftung nur bei grober Fahrlässigkeit des Verlages oder seiner Mitarbeiter in Betracht. Die Redaktion übernimmt keine Haftung für unverlangt eingesandte Manuskripte, Fotos und Illustrationen. Für Inhalte externer Seiten, auf die von Computerwoche aus gelinkt wird, übernimmt die IDG Business Media GmbH keine Verantwortung.