

# Der Entwickler 05/2001

## Nadeln im Heuhaufen - Schnittstellen und Datenbanken-Tools für Entwickler

von Henry Wolf

Hat man sich für eine Datenbank entschieden und sie zum Einsatz gebracht, beginnt oft die Suche nach kleinen Helfern für den Alltag. Dies beginnt bei der Programmierschnittstelle und setzt sich mit Tools zur Administration und datenbankgestützten Programmierung fort. Im Rahmen unserer kleinen Auswahl möchten wir Ihnen einige Tools für Interbase und Oracle näher bringen.

### 1. Datenzugriffskomponenten

Vor allem Delphi und C++-Entwickler klagen oft über die Probleme, welche bei der Softwareentwicklung gegen SQL-Datenbanken via BDE entstehen. Auch mit dbExpress können nicht alle Probleme gelöst werden. Zahlreiche Funktionalitäten ( z.B. Events, "ordentliche" request Live Query's ) einer Datenbank sind und werden auch weiterhin dem Entwickler via BDE und dbExpress verschlossen bleiben. Einige Open-Source-Entwickler und kommerzielle Unternehmen haben dies seit geraumer Zeit erkannt und bieten entsprechende native Zugriffskomponenten für Delphi – und C++Builder-Entwickler an.

Im Bereich Interbase gab es schon frühzeitig Entwickler, welche sich diesem Thema stellten. Das Ergebnis war die VCL-Komponenten-Bibliothek FreeIBKComponents. Zur selben Zeit wurden kommerzielle Bibliotheken, und da in erster Linie die Komponentenbibliothek IBQObjects, entwickelt. Diese Bibliotheken spielen heute noch eine sehr große Bedeutung. Die IB-Objects sind mittlerweile für die private Nutzung freigegeben.

Um die eigene Datenbank Interbase verstärkt zu unterstützen, integrierte Borland in Delphi/C++-Builder 5 die Klassenbibliothek IBX welche aktuell in der Version 4.6 vorliegt. Jedoch Vorsicht! Beim Ersteinsatz dieser Version traten sofort Probleme auf, für welche uns der Entwickler von IBX, Jeff Overcash, den folgenden Patch zur Verfügung stellte, den Sie Listing 1 entnehmen können:

#### Listing 1

```
function TIBCustomDataSet.PSExecuteStatement
(const ASQL: string; AParams: TParams; ResultSet:
                                     Pointer = nil): Integer;
var FQuery: TIBDataSet;
    i : Integer;
begin
  if Assigned(ResultSet) then
    begin
      TDataSet(ResultSet^) := TIBDataSet.Create(nil);
      with TIBDataSet(ResultSet^) do
        begin
          Database := Database;
          Transaction := Transaction;
          QSelect.GenerateParamNames := true;
          SelectSQL.Text := ASQL;
          for i := 0 to AParams.Count - 1 do
            Params[i].Value := AParams[i].Value;
          Open;
          Result := RowsAffected;
        end;
      end;
    end
  else
    begin
      FQuery := TIBDataSet.Create(nil);
      try
```

```

FQuery.Database := Database;
FQuery.Transaction := Transaction;
FQuery.QSelect.GenerateParamNames := True;
FQuery.SelectSQL.Text := ASQL;
for i := 0 to AParams.Count - 1 do
  FQuery.Params[i].Value := AParams[i].Value;
FQuery.ExecSQL;
if FQuery.SQLType = SQLSelect then
  Result := FQuery.Record.RecordCount
else
  Result := FQuery.RowsAffected;
finally
  FQuery.Free;
end;
end;
end;

```

Nach der Installation editieren Sie die Methode *PSExecuteStatement* der Klasse *IBCustomDataSet*. Die Quellen befinden sich im VCL-Source-Verzeichnis. Jetzt müssen Sie nur noch dafür sorgen, daß das VCL-Verzeichnis mit im Suchpfad Ihres Compilers steht, oder Sie kopieren alle Dateien *ib\*.pas* aus dem VCL-Source-Verzeichnis in ein eigenes und tragen dieses dann in den Suchpfad ein. Eine Version 4.6.1 welche diesen Patch beinhaltet ist geplant. Neben der IBX-Komponenten-Bibliothek für Delphi 4/5 existiert in Delphi 6 / Kylix eine Linux-Windows-kompatible Version, aktuell 4.5. Neben den erwähnten Komponenten-Bibliotheken existieren im Internet zahlreiche Einzelkomponenten, welche auf diesen Bibliotheken basieren. Hierfür ist die Internet- Seite <http://www.ibphoenix.com> die Quelle.

Im Bereich der nativen Schnittstellen für Delphi/C++-Builder-Entwickler sind die Lösungen der Unternehmen Core Lab Inc. ( ODAC ) und Allround Automations ( DOA ) am weitesten verbreitet. Core Lab Inc. bietet bereits sein Produkt für Delphi 6 und Kylix an. Bei DOA steht dies noch aus, ist aber in Arbeit.

Die Lösung von ODAC zeichnet sich durch einen neuen Zugriffs-Mechanismus auf entfernte Oracle-Datenbanken aus. Normal muß immer und auf jeder Maschine, von welcher aus nativ auf eine Oracle-Datenbank zugegriffen werden soll, ein Oracle- Client installiert werden. Und das ist bekanntlich immer ein Haufen Zeug. ODAC ermöglicht jetzt den Zugriff auf eine entfernte Datenbank ohne Oracle-Client, rein über TCP/IP. Mit dieser Klassenbibliothek erstellte Anwendungen benötigen für den Betrieb keinen Oracle-Clients, sondern eben nur ein installiertes IP-Protokoll. Der Zugang auf diesem Wege ist extrem schnell. Zusammen mit dem ODAC – basierenden Entwickler-Tool HORA kann man Oracle-Client freie Entwickler – und Anwenderarbeitsplätze aufbauen.

Neben den nativen Zugriffskomponenten für Delphi bietet die Firma Core Lab auch eine Crossplattform-Library für C++-Entwickler zum nativen Zugriff auf Oracle Datenbanken an. Im Rahmen der Open-Source-Lösungen errang die Komponenten-Bibliothek NCOCI8 von Dimitry Arevjev eine, unserer Meinung nach mehr als berechnete, hohe Wertschätzung. Diese Schnittstelle zeichnet sich durch eine hohe Kompatibilität in der Komponenten- Struktur zur BDE aus.

Mit dem Paket SQL-Direct wird den Entwicklern eine Lösung zum kompatiblen nativen Zugriff auf verschiedene Datenbanken zur Verfügung gestellt. Wer also gegen mehrere SQL-Ziel-Datenbanken entwickeln will, muß nicht auf die SQL-schwache BDE zurückgreifen, sondern hat mit dieser Datenbank-Programmierschnittstelle eine sehr gute Alternative.

Produkt	Webadresse / Bezugsquelle	Werkzeug / DB	Lizenz
FreeIBComponents	<a href="http://www.ibphoenix.com">http://www.ibphoenix.com</a>	Delphi 2-5, InterBase	Open Source
IBObjects	<a href="http://www.ibobjects.com/">http://www.ibobjects.com/</a>	Delphi 4-5, InterBase	Trustware
FIPlus	<a href="http://www.second-dream.ru">http://www.second-dream.ru</a>	Delphi 4-5, InterBase	Open Source
IBX	<a href="http://www.borland.com">http://www.borland.com</a>	Delphi 4-6, Kylix C++-Builder 4/5, InterBase	Open Source
SQL-Direct	<a href="http://www.sqldirect-soft.com">http://www.sqldirect-soft.com</a>	Delphi 3/4/5, C++Builder 3/4/5, Oracle 7 & 8, IBM DB2, SQL-Base, Sybase, Informix, InterBase	Kommerziell
DOA	<a href="http://www.allroundautomations">http://www.allroundautomations</a>	Delphi 2-5, Oracle 7 & 8	Kommerziell

	<a href="#">.nl</a>		
ODAC	<a href="http://www.corelab.com">http://www.corelab.com</a>	Delphi 2-6 / Kylix C++Builder 2-5, Oracle 7 & 8	Kommerziell
C++-OCL	<a href="http://www.corelab.com">http://www.corelab.com</a>	C++, Oracle 7 & 8	Kommerziell
NCOCI8	<a href="http://www.da-soft.com">http://www.da-soft.com</a>	Delphi 4 und 5, Oracle8i	Open Source

**Tabelle 1: Native Zugriffskomponenten für Datenbanken**

Durch die Integration von ADO/MDAC in das Betriebssystem Windows stellt diese Schnittstelle zum Zugriff auf Datenbanken auch für Delphi und C++-Builder-Entwickler zunehmend eine Alternative zur BDE, bzw. dbExpress und nativen Komponenten dar. ADO ist zwar in Delphi integriert, jedoch gibt es einige Anbieter, welche bessere Lösungen bieten. Dieses sind in erster Linie:

**DIAMOND ADO** <http://www.islamov.com/diamondado/>

**Kamiak ADO** <http://www.kamiak.com/>

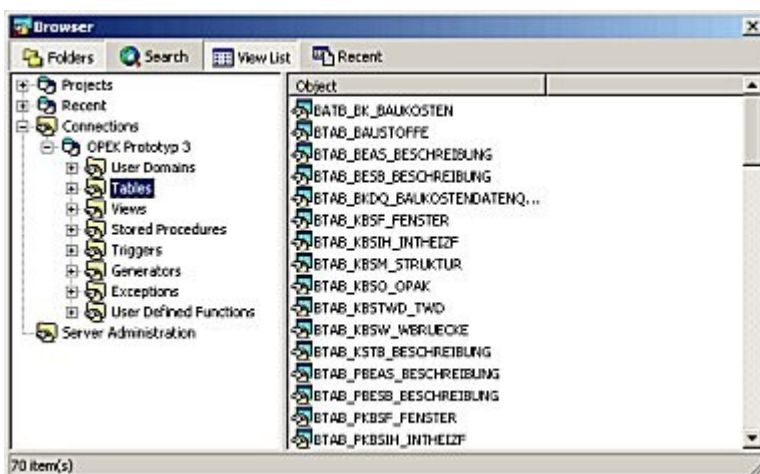
**ADONIS 4.0** <http://www.cybermagic.co.nz/adonis/>

Dies sind kommerzielle Produkte von Herstellern, welche im Bereich ADO über ein großes Know How verfügen und deren Lösungen der Delphi-eigenen überlegen sind.

## 2. Tools für Interbase

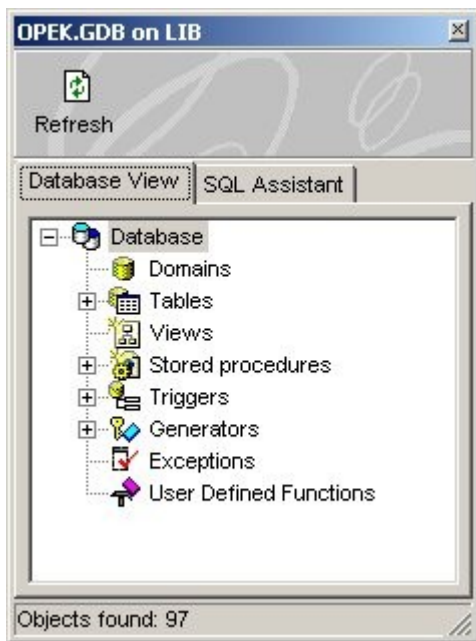
Interbase als Datenbank an sich kommt weitestgehend ohne größere Unterstützung durch Tools daher. Die Open-Source-Editionen ( Borland – Open – Source bzw. Fireball ) bieten neben der IB-Console, welche die grundlegenden Funktionen der Datenbankverwaltung ( manuelles Backup / Recovery, Nutzerverwaltung ) toolseitig unterstützt, nur noch, den seit langem bekannten SQL-Interpreter ISQL. Die kommerzielle Borland-Distribution beinhaltet neben den aus den Open-Source-Distributionen bekannten Tools noch ein Tool zur Unterstützung von einfachen Replikationen von Interbase-Datenbanken. Dieses Problem erkennend, bieten zahlreiche Entwickler aus der Open-Source Szene und kommerzielle Anbieter Tools zur Unterstützung der Administration / Verwaltung der Datenbank sowie zur Unterstützung der Softwareentwickler an.

Seit kurzer Zeit ist ein unter Interbase-Entwicklern sehr bekanntes Tool, eigentlich aus der Vergangenheit betrachtet DAS Interbase-Tool schlechthin, Marathon von Gimbel Software Service, als Open-Source Produkt im Netz erhältlich. Zum Download werden derzeit die letzte kommerzielle Version ( 1.5 ) und eine Beta-Version der neuen Open Source Version 2.0.1.31 frei zur Verfügung gestellt. Marathon ist ein Tool zur Erstellung und Bearbeitung von Interbase – Datenbankobjekten. Es verfügt über eine Debugger und einen SQL-Profiler zur Analyse von SQL-Statements und datenbankbasierenden Programmmodulen ( Stored Procedure / Trigger ).



**Abb. 1: Objekt-Baum von Marathon 2**

Auf Basis des Konzeptes von Marathon entstanden einige, vor allem kommerzielle Tools, welche das Konzept von Marathon verfeinert und erweitert haben. Dies sind in erster Linie EMS-QuickDesk und IB-Expert. Hinzugekommen ist eine umfassende Unterstützung von Interbase 6. Desweiteren können in diese Tools sogenannte Plug-Ins, sprich andere Interbase-Tools wie LogManager, IB-Compare und IB-Backup, welche wir im Anschluß noch kurz betrachten, integriert werden. So lässt sich eine effiziente Interbase-Administrations- und Entwickler-Konsole herstellen.



**Abb. 2: Objekt-Baum von EMS-Quickdesk**

Im Bereich der Freeware-Lösungen lehnt sich vor allem das Produkt IB-Admin ebenfalls sehr eng an den Funktionsumfang von Marathon u.ä. Produkten an. Die kommerziellen Produkte sind sehr effizient und stabil. Die IB-Admin ist noch nicht ganz soweit. Die aktuelle Beta-Version sieht jedoch sehr vielversprechend aus und kann in Zukunft neben Marathon den kommerziellen Lösungen Konkurrenz machen.

Auch einige sehr interessante "kleine" Helfer, welche zusätzlich als Plug-In für kommerzielle Tools oder als Einzelprodukte verfügbar sind, werden via Internet auf den relevanten Seiten zur Verfügung gestellt. Sehr hilfreich ist z.B. das Tool IB-Compare. Dies ermöglicht es, die Metadaten von 2 Datenbanken zu erstellen und ein Script zu generieren, welche die Metadaten der einen Datenbank auf den Stand der gewählten Masterdaten bringt. Hierbei können auch Datenbanken der Versionen 4 und 5 mit Datenbanken der Version 6 von Interbase verglichen werden. Es wird ein Script erzeugt, welches auch die Syntax der verschiedenen Versionen der Interbase-Datenbank berücksichtigt. Hierbei werden DDL-Scripte verglichen. Diese lassen sich sehr leicht, via ISQL erzeugen.

Adäquat zum IB Data-Pump, welcher dem Transfer von Daten aus einer BDE-Datenquelle in eine andere dient und mit Delphi Enterprise ausgeliefert wird, gibt es das Freeware-Tool IB-Pump, welches diese Funktionalität, speziell auf Interbase abgestimmt übernehmen kann. Dabei kann man Daten von einer Interbase-Datenbank zu einer anderen "pumpen" oder Daten aus einer beliebigen BDE-Datenquelle in eine Interbase-Datenbank übernehmen.

Im Bereich Replikation werden weltweit 2 relevante Produkte angeboten. Das Replikationstool der Firma Replicant Technologie wird mit der Borland – Interbase – Distribution vertrieben und ist sehr gut für einfache Replikationslösungen geeignet.

Die Replikationslösung der SDCTec GmbH zeichnet sich, neben dem in Deutschland vorhandenen Know How und Support, durch die Verfügbarkeit globaler Generatoren und die Möglichkeit der Integration mehrerer Replikations-Server in eine Netzwerkumgebung, welches sich insbesondere bei großen Netzwerk-Topologien durch eine entsprechende Toolunterstützung sehr positiv bemerkbar macht, aus.

Neben den erwähnten Tools entstanden und entstehen nach der Freigabe von Interbase als Open-Source-Datenbank sehr viele Tools, welche allerdings oft nur bereits vorhandenes wiederholen, kompakter und vielleicht auch besser. Aber etwas fehlt halt immer. Als Zugang zur Welt der vielen kleinen Interbase-Tools empfehlen wir die Seite [www.ibphoenix.com](http://www.ibphoenix.com).

Produkt	Webadresse / Bezugsquelle	Lizenz
Marathon	<a href="http://www.gimbal.com.au/">http://www.gimbal.com.au/</a>	Open Source
EMS-QuickDesk	<a href="http://www.ems-hitech.com">http://www.ems-hitech.com</a>	Version 2 Kommerziell Version 1 Freeware
IB-Expert	<a href="http://www.ibexpert.com">http://www.ibexpert.com</a> <a href="http://www.dibug.de/IBExpert/ibexpert.html">http://www.dibug.de/IBExpert/ibexpert.html</a>	Kommerziell

IB ACCESS	<a href="http://www.ibaccess.org/">http://www.ibaccess.org/</a>	Open Source
IB Admin	<a href="http://ibadmin.8m.com">http://ibadmin.8m.com</a>	Freeware
IB Backup	<a href="http://www.equitania.de">http://www.equitania.de</a>	Freeware
IB Comperer	<a href="http://www.ibphoenix.de">http://www.ibphoenix.de</a>	Freeware
IB DataPump	<a href="http://www.infogeek.de">http://www.infogeek.de</a>	Freeware
IB-Replication Demon	<a href="http://www.sdctec.net">http://www.sdctec.net</a>	Kommerziell
Replicant Technologies	<a href="http://www.replication.co.za">http://www.replication.co.za</a>	Kommerziell

**Tabelle 2: Tools für InterBase**

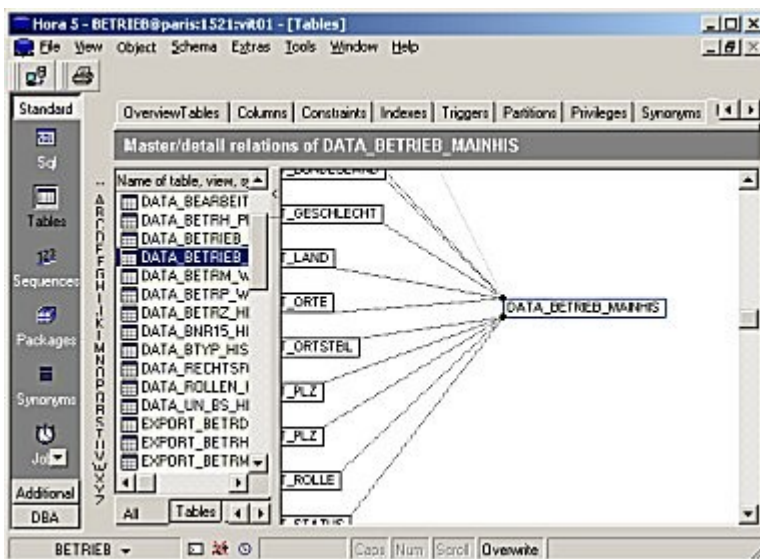
### 3. Oracle

Wer eine Oracle – Datenbank erwirbt, erhält mit der Datenbank in der Regel den Enterprise-Manager. Mit diesem Werkzeug können alle grundlegenden Datenbankverwaltungsaufgaben problemlos realisiert werden. Geht es jedoch um Problem-Lösungen im Bereich Administration und Softwareentwicklung, müssen teure Zusatz-Tools erworben werden, oder man behilft sich mit ausgeprägten Kenntnissen der Dictionary von Oracle und wertet diese via SQL aus. Prinzipielle Probleme des Enterprise-Managers und seiner Add-On-Tools sind, in der Folge der Umstellung der Entwicklung dieser Tools auf Java, die Arbeitsgeschwindigkeit und der benötigte Hauptspeicher und Plattenbedarf. Für den Administrator einer Oracle basierenden Datenbanklösung ist der Enterprise-Manager immer erste Wahl. Entwickler benötigen andere, kompakte Tools. Auch hier bietet Oracle sicherlich einiges. Wir wollen uns diesbezüglich ein paar Produkte von anderen Herstellern anschauen, welche eine Alternative darstellen.

Alle Softwarekomponenten, welche Oracle anbietet, von der Datenbank an sich, bis hin zu allen Verwaltungs- und Entwicklungstools, können von Entwicklern über das Oracle Technology Network ([technet.oracle.com](http://technet.oracle.com)) zu Testzwecken herunter geladen werden. Insbesondere Entwickler wünschen sich auch im Oracle – Umfeld, kompakte und einfach zu handhabende Tools. Neben den Tools von Oracle bieten einige weitere Tool – Hersteller gute und preiswerte Lösungen an.

Eine sehr umfassende Lösung stellt die Toolsammlung KeepTool der Berliner KeepTool GmbH dar. KeepTool in der neuen Version 5 basiert auf der aktuellen ODAC – Library und benötigt für den Oracle-Zugriff keinen Oracle-Clients mehr auf dem jeweiligen Arbeitsplatz. Eine kompakte Entwicklungsumgebung ist die positive Konsequenz.

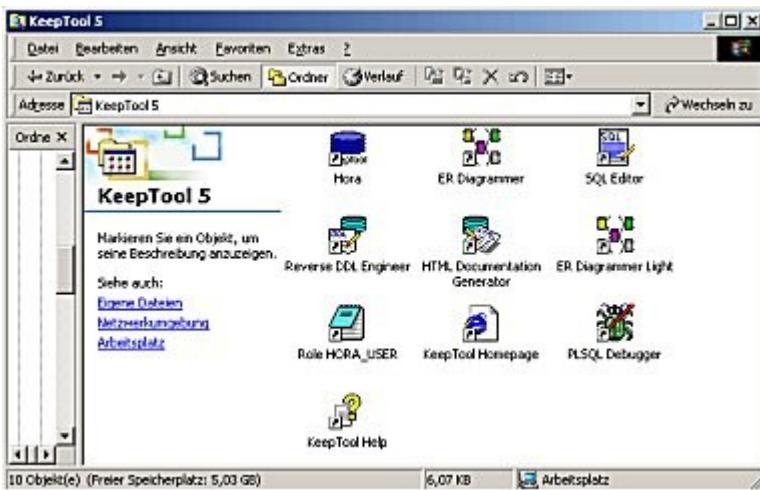
Das Paket Keep-Tool Version 5 setzt sich aus drei, auch einzeln verfügbaren, Komponenten zusammen. *HORA* ist ein Tool, welches die wichtigsten Funktionalitäten zur Administration, SQL und PL/SQL-Entwicklung und der Performance-Analyse einer Oracle-Datenbank in sehr kompakter Form zur Verfügung stellt. Der *PL/SQL-Debugger* ist ein Tool zur Fehleranalyse und zum Testen von datenbank-basierendem Programmcode. Das Add-On-Tool ER-Diagrammer ist als ein Add-Tools zu HORA ein Werkzeug zur Modellierung von Datenbanken. Hierbei können auch vorhandene Datenbanken via Reverse – Engineering grafisch dargestellt und zukünftig gepflegt werden.



**Abb. 3: Oberfläche von HORA 5**

Für Entwickler, welche einfache Werkzeuge zur Entwicklung von gespeicherten Procedures und Funktionen

bzw. Triggern suchen, bieten die holländische Firma Allround Automations und die Firma Core Lab einfache und kompakte Lösungen an. Neben reinen Editoren zur Erstellung und Compilierung von PL/SQL und SQL-Code bieten diese bei den Lösungen integrierte Debugger für gespeicherte Procedures und Funktionen.



**Abb. 4: Programmgruppe Keep-Tool Version 5**

Auf der Basis von NCOCI8 wurde von A. Kochetov ein kleines SQL-Tool entwickelt, welches über einige, sonst nicht vorhandene Funktionen verfügt. Eine besonders interessante Funktionalität dieses Tools ist ein spezieller Export von Metadaten in SQL, bei welchem für Tabellen und Indizes unterschiedliche Ziel-Tablespaces angegeben werden können. Hat man z.B. vergessen, Tabellen und Indizes beim Entwurf der Datenbank in verschiedene Tablespaces zu legen, exportiert man das Schema, gibt die gewünschten Ziel-Tablespaces an und hat somit die SQL-Skripte um das Schema neu und korrigiert anzulegen. Die Daten dann zu übernehmen, sollte kein Problem darstellen ( Delphi Daten-Pumpe u.a. ).

Produkt	Webadresse / Bezugsquelle	Lizenz
Oracle – Tools	<a href="http://technet.oracle.com">http://technet.oracle.com</a>	Kommerziell
KeepTool	<a href="http://www.keeptool.com">http://www.keeptool.com</a>	Kommerziell
PL/SQL Developer	<a href="http://www.allroundautomations.nl">http://www.allroundautomations.nl</a>	Kommerziell
SQL-Designer	<a href="http://www.corelab.com">http://www.corelab.com</a>	Kommerziell
SQL-Tools von A. Kochetov	<a href="http://www.da-soft.com">http://www.da-soft.com</a>	Freeware

**Tabelle 3: Tools für Oracle**

Auch in der Oracle-Welt gilt: Es gibt Tools wie Sand am mehr, nur manchmal kann man sich wiederholen. Für den Nutzer, welcher ein umfassendes Tool sucht, ist HORA eine sehr gute Lösung. Wer nur datenbankgestützt programmieren will, hat die Auswahl zwischen den Produkten von Core Lab und Allround Automations. Wer weiteres sucht, dem möchten wir die Internet-Adresse des Oracle-Technet ( siehe Tabelle 3 ) bzw. [www.orasearch.com](http://www.orasearch.com) empfehlen.

#### 4. Fazit

In der weiten Welt des Internets findet man Tools für fast jeden Zweck und für jeden Geldbeutel. Hierbei spielen vor allem im Bereich Interbase Open-Source-Lösungen, bzw. kommerzielle Produkte, welche auf die Open-Source-Schiene aufsetzen, eine immer größere Rolle. Auf jeden Fall lohnt es sich immer, nach kleinen Helfern Ausschau zu halten. Viele kommerzielle sind auch Ihr Geld wert und vermeiden einige der leidlichen Alltagsprobleme. Entscheidungsgrundlage, insbesondere hinsichtlich des Investitionsschutzes, für den Einsatz eines Tools sollte nicht irgendein Markenname, sondern die Durchdringung im Markt sein. Für ein Tool mit hoher Marktdurchdringung wird sich immer (-; jemand finden, der dieses pflegt und fortführt. Außerdem spricht hohe Marktdurchdringung zumindest im Entwickler-Bereich nicht immer gegen Qualität, egal ob Open Source, Freeware oder kommerzielle Lösungen.